



Segundas Jornadas Argentinas de Didáctica  
de las Ciencias de la Computación



JADiCC  
20  
22

LIBRO DE ACTAS



FACULTAD DE CIENCIAS  
EXACTAS Y NATURALES  
Y AGRIMENSURA

Fundación  
SADOSKY

<Program.AR/>



UNIVERSIDAD  
NACIONAL  
DEL NORDESTE

# Actas de las Segundas Jornadas de Didáctica de las Ciencias de la Computación (JADiCC 2022)

18, 19 y 20 de agosto de 2022

Corrientes, Argentina

Fundación Sadosky  
Facultad de Ciencias Exactas y Naturales y Agrimensura  
Universidad Nacional del Nordeste

Actas de las Segundas Jornadas Argentinas de Didáctica de las Ciencias de la Computación-JADiCC 2022 / Pablo Martínez López ... [et al.] ; compilación de Emanuel Agustín Irrazábal; Gladys Noemí Dapozo. - 1a ed compendiada. - Corrientes: Universidad Nacional del Nordeste. Facultad de Ciencias Exactas, 2022.

Libro digital, PDF

Archivo Digital: descarga y online  
ISBN 978-987-3619-81-6

1. Didáctica. 2. Computación. 3. Formación Docente. I. Martínez López, Pablo. II. Irrazábal, Emanuel Agustín, comp. III. Dapozo, Gladys Noemí, comp.  
CDD 004.0711

## Organización general

Mara Borchardt  
Gladys Dapozo  
Emanuel Irrazábal

## Chairs

Emanuel Irrazábal (UNNE), Gladys Dapozo (UNNE), Christian Cossio-Mercado (UBA)

## Comité de programa

Araceli N. Acosta (UNC)	Marcos Gómez (Fundación Sadosky)
Alfredo Héctor Sanzo (UTN/UNQui)	Maria Carmen Leonardi (UNCPBA)
Ana María Company (UNNE)	María Fernanda Golobisky (UTN-FRSF)
Claudia Banchoff (UNLP)	Marta Lasso (UNPA)
Claudia Casariego (UNLa)	Matías López-Rosenfeld (UBA-CONICET)
Claudia Deco (UNR)	Natalia Monjelat (IRICE -CONICET-UNR)
Emmanuel Frati (UNDEC)	Noelia Soledad Pinto (UTN FRRe)
Fernando Puricelli (UNAHUR)	Nora Reyes (UNSL)
Flavia Millán (UNSI)	Pablo E. Martínez López (UNQui)
Francisco Bavera (UNRC)	Pablo Turjanski (UBA-CONICET)
Jacqueline Fernández (UNSL)	Silvia Villagra (UNPA-UACO)
Jorge Rodríguez (UNComa)	Sonia I. Mariño (UNNE)
Laura Virginia Garbarini (UNLa)	Verónica Andrea Bollati (UTN FRRe CONICET)
Lucila Romero (UNL)	
Marcela Daniele (UNRC)	

### Sitio web

Iván Sambrana

### Diseño gráfico

Agustina Acosta

### Comunicación

Ana Belén Cavalieri

## Autoridades FaCENA-UNNE

Decana: Mgter. María Viviana Godoy Guglielmone

Vicedecano: Dr. Enrique Rafael Laffont

## Comité Organizador FaCENA-UNNE

Gladys Dapozo (Coordinadora)

Directora carrera Licenciatura en Sistemas de Información

María de los Ángeles Ferraro

Directora Departamento de Informática

Raquel Petris

Coordinadora Convenio Sadosky-UNNE

Yanina Medina

Coordinadora Diplomatura en Programación y Robótica Educativa

Beatriz Castro Chans

Responsable del Área de Educación Virtual de FaCENA (UNNE)

Emanuel Irrázabal

Director proyecto de investigación en Ciencias de la Computación

Sonia Mariño

Directora proyecto de investigación en Ciencias de la Computación



## Índice

Introducción .....	8
Conferencias.....	9
“Problemática de la participación de mujeres y diversidades en el ámbito educativo o laboral vinculado a las Ciencias de la Computación (CC)” .....	9
“Vuela bajo porque abajo, está la verdad" (el hardware y sistema operativo).....	9
Mesas debate .....	10
“Propuesta curricular para la inclusión de Ciencias de la computación en el nivel primario y secundario de Argentina” .....	10
“Qué y cómo enseñar CC en las Universidades” .....	10
“Presentación de la propuesta de Educación Tecnológica de PLaNEA – UNICEF”. .....	10
Talleres .....	11
Taller 1. Qué y Cómo Enseñar Ciencias de la Computación.....	11
Taller 2. Introducción a infraestructura para aplicaciones IoT .....	11
Taller 3. La pregunta del millón: los qué y los cómo de la introducción de las CC en la escuela.....	11
Taller 4. Creando mundos, abriendo historias. Acercamiento a las narrativas interactivas desde una mirada ampliada.....	11
Taller 5. Programación y Animación con p5.js.....	11
Taller 6. Un primer acercamiento al aprendizaje automático y sus usos. ....	12
Taller 7. Orientando a Objetos con Diseño Audiovisual .....	12
Taller 8. Taller de vocaciones: Seguridad Informática .....	12
Taller 9. Taller de vocaciones: Ciencias de Datos.....	12
Taller 10. ¿Aprenden las máquinas? Aportes desde las neurociencias a la IA aplicados a espacios educativos.....	12
Taller 11. Relación entre la enseñanza de las Ciencias de la Computación y la brecha de género. ....	13
Taller 12. Inteligencia Artificial: datos, sesgos, criterios y personas.....	13
Presentación de Posters.....	14
Experiencia de Formación Docente Continua en Corrientes: Diplomatura en Programación y Robótica Educativa. <i>Sofía Vallejos, Darío Báez, Claudia Giménez</i> . I.S.F.D. José M. Estrada... ..	14
Nuevas formas de evaluar Ciencias de la Computación. <i>Melina Belén Galán, Edgardo David Alegre</i> . UNIPE .....	14
Programame: nuevos caminos en la formación docente. <i>Liliana Patricia Arbelo, Georgina Alejandra Oliva Morales</i> . Instituto Superior de Formación Docente y técnica "Curuzú Cuatiá .....	14
Proyecto Semáforos de mi ciudad. <i>Carolina Wayar</i> . I.S.F.D. José M. Estrada .....	14
Nuestra experiencia como docente remoto del Plan Ceibal. <i>Graciela Susana Gómez, Nancy Elizabet Rivero</i> . Fundación Sadosky .....	14

Propuesta didáctica para la enseñanza de programación a docentes en formación. <i>Silvina Elena Busto</i> . ISFD N° 50 Berazategui, Buenos Aires. ....	14
Woblocks: Wollok Game + Blockly. <i>Alejandro Ferrante, Matías López Y Rosenfeld, Nahuel Palumbo, Alfredo Héctor Sanzo</i> . UBA, CONICET .....	14
Herramientas de autor en la construcción de objetos de aprendizaje. <i>Paula Lorena López</i> . ISP Dr. Joaquín V. González.....	14
Enseñanza-Aprendizaje de Algoritmos y Estructuras de Datos en Proyectos Integradores mediante Metodología ABP. Oscar Adolfo Vallejos, Jaquelina Edit Escalante, Mariela Burghardt UNNE.....	14
Formación en Accesibilidad Web. <i>Sonia I. Mariño, Pedro L. Alfonso, Verónica K. Pagnoni, Maria L. Gronda, Jose M. Bordon</i> UNNE .....	14
Pasantías educativas en un área TIC de gobierno. Estrategia de formación en Ingeniería del Software. <i>Sonia I Mariño, Maria de Los Ángeles Ferraro, Silvana Armana, Laura I. Gómez Solís, Yanina Medina</i> . UNNE.....	14
Experiencia Docente en tiempos de pandemia: Cómo innovar en la enseñanza universitaria utilizando modalidad híbrida. <i>Mirtha Giovannini, Noelia Soledad Pinto, Nicolas Tortosa</i> . UTN-FRRE.....	14
Formación Docente en Ciencias de la Computación para la Educación y el Trabajo del Futuro. Francisco Bavera, Marcela Daniele, Teresa Quintero. UNRC.....	14
Estrategias híbridas mediadas por una pizarra digital interactiva. Una experiencia en la asignatura Proyecto Final de Carrera. <i>Sonia I. Mariño, Paola E. Insaurralde, Romina Y. Alderete, Marcelo O. Cáceres</i> . UNNE .....	14
Una Estrategia Didáctica Áulica para la Enseñanza de Ingeniería de Requerimientos. <i>Ezequiel Moyano, Marcela Jerez, Martin Villarreal, Romano Lucas, Thomas Villanueva, Brian Rigoni, Daniel Lost</i> . UNTDF.....	14
Propuesta de un clasificador según problemas comunes en la gestión de proyectos de desarrollo de software en ambientes de práctica profesional supervisada. <i>Alice R. Rambo, Mariana Boari, Roberto L. Sueldo, Mirian I. Rodríguez</i> . UNaM .....	14
Presentación de Artículos académicos .....	15
La TGS como base para el pensamiento computacional en la formación de profesionales informáticos. Autores: <i>Flavia Moreira, Beatriz Castro Chans, María Lorena Guastavino Mosna y Guillermo Andrés Arduino</i> . ....	17
Educación Mediada en Primer Año de Ingeniería para la Formación de Competencias de la Industria 4.0 en el Contexto de la Enseñanza Remota de Emergencia. Autores: <i>Artemisa Trigueros, Mabel Compagnoni y Larisa Toro</i> . ....	26
Gestión del Tiempo en Actividades Remotas de Estudiantes Ingresantes de la Carrera Lic. en Sistemas de la FaCENA – UNNE. Autores: <i>María Ornella Del Grosso, María Del Mar Mosquera Mosquera and Mirta Fernández</i> . ....	36
Evaluación de habilidades de pensamiento computacional al inicio de una asignatura de programación en una carrera de Informática. Autores: <i>Gladys Dapozo, Cristina Greiner, Raquel Petris, Emanuel Irrazabal, Ana María Company, María Cecilia Espíndola y Yanina Medina</i> . ....	49

Formación de Promotores en Seguridad Informática para el ámbito escolar. <i>Autores: Javier Agustín Ferreira, Cecilia Martínez y Nicolás Wolovick.</i> .....	59
Dictado del curso “La Programación y su Didáctica 2” en FaCENA – UNNE. <i>Autores: Ana María Company, María Cecilia Espíndola, Raquel Petris, Numa Badaracco y Gladys Dapozo.</i> .....	71
Taller de Robótica Educativa en Nivel Inicial: una experiencia de formación docente. <i>Autores: Gabriela Cenich, Laura Felice, Andrea Miranda, María Carmen Leonardi y María Virginia Mauco.</i> .....	76
Lógica, Programación y su Enseñanza: Un curso introductorio para la formación docente. <i>Autores: Araceli N. Acosta y Cecilia Martínez.</i> .....	86
Desde los cimientos: Un estudio sobre la formación inicial del profesorado en relación a la capacitación docente continua en el área de la programación. <i>Autora: Silvia Pilar Rodríguez</i> .....	98
MAPUDUNGUN 2.0. <i>Autora: Marcia Gabriela Pizarro.</i> .....	111
El ABP en el redictado de materias de programación: una experiencia en la educación superior. <i>Autores: Natalia Colussi, Natalia Monjelat y Pamela Viale.</i> .....	120
Aprender programación usando bloques y texto en forma simultánea - Un enfoque espiralado. <i>Autores: Gonzalo Fernández, Pablo Martínez López y María Cecilia Martínez.</i> .....	130
Hay piratas en la sala de informática. <i>Autora: Graciela Ivana Guzmán.</i> .....	139
Keylogging y screencapture como herramientas de laboratorio para estudios de casos en ciencias de la computación. <i>Autores: Valentin Basel.</i> .....	142
Evaluación de calidad de Entornos de Desarrollo Integrado de Robótica Educativa. Artículos originales. <i>Autores: Martín Mariano Lobos, Silvia Gabriela Bast y Gustavo Javier Astudillo.</i> ..	15
Taller de Programación de apps para celulares. Generando vocaciones científicas en los jóvenes estudiantes de nivel secundario. <i>Autores: Ariel Ferreira Szpiniak, Ivana Cruz, Lucía Antuña y Carolina Borbolla.</i> .....	168
“Somos muy pocas mujeres”. Diferencias de Género en carreras de Computación en los casos de UNC y UNComa y la búsqueda de referentes mujeres en el área. <i>Autores: María Emilia Echeveste, Jorge Rodríguez, Marcos Gómez, Marina Moran y Luciana Benotti.</i> .....	180
La incorporación de las Ciencias de la Computación en el curriculum: ¿la transversalidad como alternativa? <i>Autora: Magdalena Garzón.</i> .....	195
Piche: evolución del robot educativo soberano. <i>Autores: Ariel Ferreira Szpiniak, Carlos Maximiliano Correa, Pablo Leonel Etcheverry y Carlos Etcheverry.</i> .....	209
La Escuela de Robótica de Misiones como modelo de implementación de metodologías disruptivas de enseñanza y aprendizaje. <i>Autores: Solange Schelske, Sebastian Rossi y Marcelo Benítez.</i> .....	220
Didáctica de la Programación en Educación Inicial: la co-construcción entre estudiantes y docentes. <i>Autores: Cecilia Haydée Exeni</i> .....	233
De la basura electrónica a juegos. <i>Autores: Claudio Javier Di Paolo.</i> .....	243
Identificación de sensores utilizados en robótica educativa mediante aprendizaje automático. <i>Autores: Javier Berger y Solange Schelske.</i> .....	253

## Introducción

Estas jornadas, propiciadas por la Iniciativa Program.AR de la Fundación Sadosky, tienen la finalidad de favorecer el encuentro de investigadores, investigadoras, docentes, estudiantes, funcionarios y funcionarias públicos, que lleven adelante proyectos de investigación o de intervención educativa en la enseñanza de las Ciencias de la Computación.

Se realizan desde el 2018 buscando promover el desarrollo de las Ciencias de la Computación en todos los niveles del sistema educativo argentino. Inicialmente denominadas Jornadas de Didáctica de la Programación (JaDiPro), desde el año 2021 adoptaron la denominación de Jornadas Argentinas de Didáctica de las Ciencias de la Computación (JADiCC) para abarcar otros contenidos de la disciplina, tales como los referidos a las arquitecturas de las computadoras, el funcionamiento de las redes informáticas y la inteligencia artificial, con el objetivo de que la incorporación de las Ciencias de la Computación en las distintas instancias educativas se lleve a cabo desde una perspectiva disciplinar amplia.

La primera edición de JADiCC, organizada por la Fundación Sadosky, se realizó en modalidad virtual dadas las restricciones impuestas por la pandemia por Covid19. En esta segunda edición se buscó volver a las actividades presenciales, al encuentro de docentes y estudiantes, a las charlas de café, al debate cara a cara, aprovechando, sin embargo, las ventajas de la virtualidad para contar con la participación a distancia de destacados especialistas para el desarrollo de las conferencias programadas, que estuvieron a disposición de todos los interesados mediante su difusión por Youtube.

En este marco se desarrollaron diversas actividades, conferencias, mesas debate, talleres, presentación de posters y presentación de artículos sobre experiencias docentes y trabajos de investigación.

Se realizaron 2 conferencias, 3 mesas debates, 12 talleres (2 de ellos destinados a estudiantes de escuelas secundarias se replicaron en 2 turnos, mañana y tarde), se presentaron 16 posters y 23 artículos.

En las distintas actividades realizadas, se ha compartido y debatido sobre estrategias didácticas para la enseñanza de los temas propios de la disciplina en las escuelas y en las universidades, se han compartido experiencias y propuestas de diseños curriculares con la participación de funcionarios de educación de las provincias de Corrientes, Chaco y Misiones, se ha reflexionado sobre la problemática de género y diversidades, se ha capacitado sobre temas de seguridad Informática y ciencia de datos a los estudiantes de las escuelas secundarias.

Durante los 3 días de las jornadas participaron 580 personas, docentes, investigadores y estudiantes del nivel secundario y terciario.

Se destaca la amplia participación de universidades nacionales del país, como así también, de otras instituciones educativas de los niveles preuniversitarios y del ámbito privado. El intercambio de experiencias, saberes y reflexiones sobre problemáticas muy vigentes en el paradigma de la educación en la sociedad actual ha sido sumamente fructífero. ¡Una excelente experiencia para la UNNE y su región de influencia!

## Conferencias

### **“Problemática de la participación de mujeres y diversidades en el ámbito educativo o laboral vinculado a las Ciencias de la Computación (CC)”.**

Resumen: Se presentaron los resultados de diferentes investigaciones que abordan la problemática en cuestión, moderado por la socióloga Verónica Marino, consultora de la Fundación Quántitas, líder de una investigación en curso sobre trayectorias universitarias en carreras de Ciencias de la Computación en Argentina.

Participantes:

- **Emilia Echeveste:** es Doctora en Ciencias de la Educación, postdoctoranda CONICET con lugar de trabajo en FaMAF-UNC. Actualmente investiga la relación con el conocimiento en aprendizajes escolares en Ciencias de la Computación en Córdoba y compartirá cómo apareció en el contexto de sus investigaciones la preocupación por la perspectiva de género.
- **Ana Rapoport.** Es Magister en Políticas Públicas y coordina el área de Equidad de Género del Instituto Nacional de Educación Técnica y ha integrado previamente el área de evaluación de la institución. Compartirá con nosotros datos referidos a la participación de mujeres y diversidades en la formación secundaria técnica de todo el país y la situación de las escuelas de informática en este contexto.
- **Verónica Millenaar.** Es Doctora en Ciencias Sociales, investigadora de CONICET con sede en el PREJET-CIS/IDES. Actualmente trabaja en los enfoques de género en la formación profesional: concepciones e intervenciones institucionales. Comentaré sobre el caso particular de la formación profesional en computación y los resultados surgidos de su investigación "Mujeres en Programación: entre la reproducción y las nuevas construcciones de género. El caso de la formación en el nivel medio en la Ciudad de Buenos Aires".
- **Ivana Bachmanosvky.** Es Prof. de Ciencias Antropológicas (Universidad de Buenos Aires), becaria doctoral del CONICET con sede en el PREJET-CIS/IDES. Comentaré su investigación en relación a la agrupación Women in Games Argentina, que se define como "agrupación de mujeres y disidencias del rubro de videojuegos".

Destinatarios: Docentes de todos los niveles, público interesado en la temática.

### **“Vuela bajo porque abajo, está la verdad" (el hardware y sistema operativo).**

Disertante: **Dr. Nicolas Wolovick (UNC).** Es profesor de computación que hace algunas otras cosas como HPC, reciclado de máquinas y retro-computación. Actualmente trabaja en FaMAF-UNC y CCAD-UNC. Es también un orgulloso CyberCiruja.

Resumen: La computadora es una lasaña muy alta, capa de abstracción sobre capa de abstracción. Usualmente, en los enfoques curriculares para CC de nivel inicial, primario y secundario, las capas bajas, cómo el hardware y el sistema operativo se dejan de lado por prejuicios respecto a la dificultad. El mismo está presente también en la educación superior en CC. El conferencista tratará de argumentar que las capas bajas, por un lado, contienen todas las ideas, conceptos y métodos que definen a la Computación como una Ciencia, y por otro, que al ser más concretas al punto que se pueden tocar, bajan un poco el salto de abstracción que las y los estudiantes tienen que hacer, y con esto tal vez se mejora la transferencia de los contenidos y la aceptación por parte de ellas y ellos. Se espera un ida y vuelta de la concurrencia para debatir estas ideas.

Destinatarios: Docentes de todos los niveles, público interesado en la temática.

## Mesas debate

### “Propuesta curricular para la inclusión de Ciencias de la computación en el nivel primario y secundario de Argentina”

Moderador: Lic. Gabriel Scarano. Responsable de la Formación Docente en la Iniciativa Program.AR de la Fundación Sadosky.

Resumen: Con la participación de funcionarios de educación de la región, la directora de la Iniciativa Program.AR presentará junto a especialistas de su equipo, la propuesta curricular para la inclusión de Ciencias de la computación en el nivel primario y secundario de Argentina: definición de ejes, secuenciación de contenidos y habilidades por ciclo y nivel. Los funcionarios aportarán su visión y comentarios, desde la perspectiva de la realidad de cada provincia.

Destinatarios: Directivos y docentes de instituciones educativas de la región.

### “Qué y cómo enseñar CC en las Universidades”

Moderadoras: Mgter. Cristina Greiner, Lic. Yanina Medina (UNNE)

Resumen: Con la participación de docentes representantes de las carreras de Informática, se debatirá sobre qué y cómo enseñar CC en las Universidades: aprendizajes vinculados a la experiencia iniciada con la Programación y su didáctica y sobre la inclusión de habilidades transversales en la formación de informáticos en los planes de estudio.

Destinatarios: Docentes universitarios.

### “Presentación de la propuesta de Educación Tecnológica de PLaNEA – UNICEF”.

Moderador: Gustavo del Dago. Coordinador proyecto PLaNEA UNICEF.

Resumen: PLaNEA se trata de una línea de acción de UNICEF destinada a fortalecer la formación y el acompañamiento a docentes y equipos directivos en la transformación de la escuela secundaria, para lograr que las y los adolescentes accedan a los aprendizajes claves para su vida y completen el nivel secundario. Responsables de la implementación del programa en conjunto con responsables de las jurisdicciones de Chaco y Tucumán junto a docentes y el autor de contenidos y formación de Educación Tecnológica para PLaNEA, presentarán las propuestas pedagógicas y la experiencia de su implementación en Chaco y Tucumán.

Destinatarios: Directivos y docentes de instituciones educativas de la región.

## Talleres

### Taller 1. Qué y Cómo Enseñar Ciencias de la Computación

Docentes: Raquel Petris, Numa Badaracco; Ana M. Company, María C. Espíndola (UNNE)

Resumen: Este taller tiene como propósito realizar un recorrido por los Núcleos de Aprendizajes Prioritarios (NAP) de Educación Digital, Programación y Robótica, reflexionar y debatir acerca de los conceptos fundamentales asociados al Pensamiento Computacional que se deben abordar, y finalmente, presentar un método de resolución de problemas basado en la propuesta de Program.ar y algunas herramientas lúdicas para incorporar el pensamiento computacional en las escuelas.

Destinatarios: Docentes de niveles preuniversitarios. Estudiantes de profesorado interesados en la temática.

### Taller 2. Introducción a infraestructura para aplicaciones IoT

Docentes: Iván Sambrana, José Manuel Alonso (UNNE)

Resumen: Orientado para que los docentes comprendan la forma en que los dispositivos intercambian información y que logren crear un servicio de suscripción/publicación de mensajes obtenidos de una red de sensores y/o servicio web. Durante el taller los cursantes realizarán en forma grupal un proyecto IOT colaborativo que integre los conceptos dados y que estén aplicados a solucionar una problemática particular, que será evaluado mediante una lista de cotejo.

Destinatarios: Docentes de nivel primario, secundario o terciario.

### Taller 3. La pregunta del millón: los qué y los cómo de la introducción de las CC en la escuela.

Dictantes: Juan Gabriel Scarano, Cingolani Débora Marcela (Equipo Fundación Sadosky)

Resumen: Explorando contenidos de Ciencias de la Computación, los directivos analizarán la situación actual de sus instituciones en relación con estos saberes y formularán posibles acciones concretas y posibles de realizar en un periodo de tiempo determinado para poder incorporar de forma eficiente y contextualizada contenidos de Ciencias de la Computación en sus instituciones.

Destinatarios: Equipos de gestión de instituciones educativas.

### Taller 4. Creando mundos, abriendo historias. Acercamiento a las narrativas interactivas desde una mirada ampliada.

Dictantes: Julieta Lombardelli, Lujan Oulton, Ulla Wester (Instituto Goethe)

Resumen: Se enmarca en el proyecto SheroesInGames - Chicas como diseñadoras de videojuegos- cuyo objetivo es contribuir a que más mujeres y niñas se entusiasmen con el desarrollo de videojuegos y animarlas a hacerlo desde una edad más temprana. Se propone el uso de un Kit de herramientas para lograr este propósito. Al finalizar el taller se espera desarrollar entre las participantes un miniprototipo de videojuego que pueda ser publicado online.

Destinatarios: Docentes o personas capacitadoras.

### Taller 5. Programación y Animación con p5.js

Dictante: Chaves Maximiliano Emmanuel (Escuelas Proa. Sede Río Cuarto)

Resumen: La temática abordó la introducción a la programación y la algorítmica mediante el entorno p5.js de Processing. Se realizaron distintas actividades para introducir los conceptos de

la programación estructurada de forma visual e interactiva, con situaciones problemáticas que propongan crear animaciones, juegos y dibujos.

Destinatarios: Docentes, docentes en formación, estudiantes de nivel medio, y público en general interesado en la temática. No es necesario tener ningún conocimiento previo en programación.

### **Taller 6. Un primer acercamiento al aprendizaje automático y sus usos.**

Dictantes: Isabel Kimura, Agustín Cao, Antonella Garea, Benjamín Freccero, Paula Marcón y Claudia Queiruga (UNLP)

Resumen: Se busca introducir a las personas cursantes en las diferentes técnicas de inteligencia artificial que se utilizan en aplicaciones de uso cada vez más cotidiano. La intención es analizar y construir aplicaciones móviles que usan técnicas de aprendizaje automático (machine learning) y aprendizaje profundo (deep learning) para reconocer y crear imágenes y sonidos.

Destinatarios: Estudiantes secundario con conocimientos básicos de programación y algorítmica.

### **Taller 7. Orientando a Objetos con Diseño Audiovisual**

Dictante: Juan Furci (UNLP-UBA)

Resumen: Introduce a los cursantes en los conceptos de la programación orientada a objeto, haciendo uso de la disponibilidad de lenguajes y librerías que permiten una cuasi inmediata correspondencia entre la modificación del código e imágenes interactivas. Para ellos se utilizará la librería p5.js que corre sobre javascript junto a visual studio code, y para una versión offline, el entorno de desarrollo java-processing).

Destinatarios: Docentes y estudiantes universitarios.

### **Taller 8. Taller de vocaciones: Seguridad Informática**

Dictantes: Tomás Caballero y Fernando Cáceres (Equipo Fundación Sadosky)

Resumen: Presenta ideas básicas de Seguridad Informática a estudiantes sin ninguna formación previa en Ciencias de la Computación: Lo único que se requiere es que dispongan de un teléfono celular con Android para descargar algunas de las aplicaciones que se utilizarán para las actividades programadas.

Destinatarios: Estudiantes de los últimos dos años de secundaria

### **Taller 9. Taller de vocaciones: Ciencias de Datos**

Dictantes: Julián Dabbah y Alfredo Sanzo (Equipo Fundación Sadosky)

Resumen: El objetivo es involucrar a las y los estudiantes en una experiencia similar al análisis y procesamiento de datos

Destinatarios: Estudiantes de los últimos dos años de secundaria

### **Taller 10. ¿Aprenden las máquinas? Aportes desde las neurociencias a la IA aplicados a espacios educativos.**

Dictantes: Carina Mellibovsky, Julieta Lombardelli, Pablo Galli, Lucia Brunand (NEA- La Plata)

Resumen: Su objetivo es acercar los conceptos de las neurociencias, puestas en relación con el pensamiento computacional y el desarrollo de la Inteligencia Artificial (IA). Este espacio proporcionará a los participantes conocimientos y habilidades sobre modularización y pensamiento computacional, así como recursos para desarrollar una propuesta en el aula. Los



docentes aprenderán los principios de la programación por bloques a partir del uso de la placa micro:bit, que serán provistas en el taller.

Destinatarios: Docentes de 2do ciclo de educación primaria (cupo limitado).

### **Taller 11. Relación entre la enseñanza de las Ciencias de la Computación y la brecha de género.**

Dictante: Mariana Labhart, Laura Ramírez Rivillas, Magdalena Garzón (Equipo Fundación Sadosky)

Resumen: Aborda la importancia de la incorporación de contenidos de Ciencias de la Computación (CC) para reducir la brecha de género; y -estrategias para abordar la enseñanza de las CC desde una perspectiva de género.

Destinatarios: Docentes de todos los niveles

### **Taller 12. Inteligencia Artificial: datos, sesgos, criterios y personas.**

Dictante: Julián Dabbah, Marcos J. Gómez (Equipo Fundación Sadosky)

Resumen: Tiene como objetivos contextualizar el uso de la IA en experiencias cotidianas y reconocer el impacto de los criterios de las personas en el funcionamiento de los algoritmos de aprendizaje automático al realizar la predicción, experimentar el entrenamiento de un modelo de aprendizaje automático, asociar las predicciones de un modelo con los datos con los que se lo entrenó e identificar los problemas derivados de los sesgos en el entrenamiento de algoritmos de aprendizaje automático.

Destinatarios: Docentes de nivel primario y secundario.

## Presentación de Posters

### 1. Enseñanza de las CC en los niveles preuniversitarios

Moderadoras: Lic. María Cecilia Espíndola, Lic. Silvana Armana (UNNE)

Experiencia de Formación Docente Continua en Corrientes: Diplomatura en Programación y Robótica Educativa. *Sofía Vallejos, Darío Báez, Claudia Giménez*. I.S.F.D. José M. Estrada.

Nuevas formas de evaluar Ciencias de la Computación. *Melina Belén Galán, Edgardo David Alegre*. UNIPE

Programame: nuevos caminos en la formación docente. *Liliana Patricia Arbelo, Georgina Alejandra Oliva Morales*. Instituto Superior de Formación Docente y técnica "Curuzú Cuatía

Proyecto Semáforos de mi ciudad. *Carolina Wayar*. I.S.F.D. José M. Estrada

Nuestra experiencia como docente remoto del Plan Ceibal. *Graciela Susana Gómez, Nancy Elizabet Rivero*. Fundación Sadosky

Propuesta didáctica para la enseñanza de programación a docentes en formación. *Silvina Elena Busto*. ISFD N° 50 Berazategui, Buenos Aires.

Woblocks: Wollok Game + Blockly. *Alejandro Ferrante, Matías López Y Rosenfeld, Nahuel Palumbo, Alfredo Héctor Sanzo*. UBA, CONICET

Herramientas de autor en la construcción de objetos de aprendizaje. *Paula Lorena López*. ISP Dr. Joaquín V. González.

### 2. Enseñanza de las CC en la universidad

Moderadores: Dra. Sonia Mariño, Mgter. Julio Acosta (UNNE)

Enseñanza-Aprendizaje de Algoritmos y Estructuras de Datos en Proyectos Integradores mediante Metodología ABP. Oscar Adolfo Vallejos, Jaquelina Edit Escalante, Mariela Burghardt UNNE

Formación en Accesibilidad Web. *Sonia I. Mariño, Pedro L. Alfonso, Verónica K. Pagnoni, María L. Gronda, Jose M. Bordon* UNNE

Pasantías educativas en un área TIC de gobierno. Estrategia de formación en Ingeniería del Software. *Sonia I Mariño, María de Los Ángeles Ferraro, Silvana Armana, Laura I. Gómez Solis, Yanina Medina*. UNNE

Experiencia Docente en tiempos de pandemia: Cómo innovar en la enseñanza universitaria utilizando modalidad híbrida. *Mirtha Giovannini, Noelia Soledad Pinto, Nicolas Tortosa*. UTN-FRRE

Formación Docente en Ciencias de la Computación para la Educación y el Trabajo del Futuro. Francisco Bavera, Marcela Daniele, Teresa Quintero. UNRC

Estrategias híbridas mediadas por una pizarra digital interactiva. Una experiencia en la asignatura Proyecto Final de Carrera. *Sonia I. Mariño, Paola E. Insaurralde, Romina Y. Alderete, Marcelo O. Cáceres*. UNNE

Una Estrategia Didáctica Áulica para la Enseñanza de Ingeniería de Requerimientos. *Ezequiel Moyano, Marcela Jerez, Martín Villarreal, Romano Lucas, Thomas Villanueva, Brian Rigoni, Daniel Lost*. UNTDF

Propuesta de un clasificador según problemas comunes en la gestión de proyectos de desarrollo de software en ambientes de práctica profesional supervisada. *Alice R. Rambo, Mariana Boari, Roberto L. Sueldo, Mirian I. Rodríguez*. UNaM

## Presentación de Artículos académicos

### Track 1: Enseñanza de CC en la Universidad

Moderadores: Prof. Alfredo Sanzo (UBA), Dra. Sonia Mariño (UNNE)

La TGS como base para el pensamiento computacional en la formación de profesionales informáticos. Autores: *Flavia Moreira, Beatriz Castro Chans, María Lorena Guastavino Mosna y Guillermo Andrés Arduino.*

Educación Mediada en Primer Año de Ingeniería para la Formación de Competencias de la Industria 4.0 en el Contexto de la Enseñanza Remota de Emergencia. Autores: *Artemisa Trigueros, Mabel Compagnoni y Larisa Toro.*

Gestión del Tiempo en Actividades Remotas de Estudiantes Ingresantes de la Carrera Lic. en Sistemas de la FaCENA – UNNE. Autores: *María Ornella Del Grosso, María Del Mar Mosquera Mosquera and Mirta Fernández.*

Evaluación de habilidades de pensamiento computacional al inicio de una asignatura de programación en una carrera de Informática. Autores: *Gladys Dapozo, Cristina Greiner, Raquel Petris, Emanuel Irrazabal, Ana María Company, María Cecilia Espíndola y Yanina Medina.*

### Track 2: Estrategias y Enseñanza/ Formación docente/Enseñanza de la programación

Moderadores: Dr. Pablo Martínez López (UNQui), Mgter. Raquel Petris (UNNE)

Formación de Promotores en Seguridad Informática para el ámbito escolar. Autores: *Javier Agustín Ferreira, Cecilia Martínez y Nicolás Wolovick.*

Dictado del curso “La Programación y su Didáctica 2” en FaCENA – UNNE. Autores: *Ana María Company, María Cecilia Espíndola, Raquel Petris, Numa Badaracco y Gladys Dapozo.*

Taller de Robótica Educativa en Nivel Inicial: una experiencia de formación docente. Autores: *Gabriela Cenich, Laura Felice, Andrea Miranda, María Carmen Leonardi y María Virginia Mauco.*

Lógica, Programación y su Enseñanza: Un curso introductorio para la formación docente. Autores: *Araceli N. Acosta y Cecilia Martínez.*

Desde los cimientos: Un estudio sobre la formación inicial del profesorado en relación a la capacitación docente continua en el área de la programación. Autora: *Silvia Pilar Rodríguez*

### Track 3: Enseñanza de la programación

Moderadores: Prof. Christian Cossio (UBA), Mgter. Cristina Greiner (UNNE)

MAPUDUNGUN 2.0. Autora: Marcia Gabriela Pizarro.

El ABP en el redictado de materias de programación: una experiencia en la educación superior. Autores: Natalia Colussi, Natalia Monjelat y Pamela Viale.

Aprender programación usando bloques y texto en forma simultánea - Un enfoque espiralado. Autores: Gonzalo Fernández, Pablo Martínez López y María Cecilia Martínez.

Hay piratas en la sala de informática. Autora: Graciela Ivana Guzmán.

Keylogging y screenshot como herramientas de laboratorio para estudios de casos en ciencias de la computación. Autores: Valentin Basel.

Evaluación de calidad de Entornos de Desarrollo Integrado de Robótica Educativa. Artículos originales. Autores: Martín Mariano Lobos, Silvia Gabriela Bast y Gustavo Javier Astudillo.

#### Track 4 - Computación y sociedad

Moderadores: Mg. Laura Felice (UNCPBA), Dr. Emanuel Irrazábal (UNNE)

Taller de Programación de apps para celulares. Generando vocaciones científicas en los jóvenes estudiantes de nivel secundario. *Autores: Ariel Ferreira Szpiniak, Ivana Cruz, Lucía Antuña y Carolina Borbolla.*

“Somos muy pocas mujeres”. Diferencias de Género en carreras de Computación en los casos de UNC y UNComa y la búsqueda de referentes mujeres en el área. *Autores: María Emilia Echeveste, Jorge Rodríguez, Marcos Gómez, Marina Moran y Luciana Benotti.*

La incorporación de las Ciencias de la Computación en el curriculum: ¿la transversalidad como alternativa? *Autora: Magdalena Garzón.*

Piche: evolución del robot educativo soberano. *Autores: Ariel Ferreira Szpiniak, Carlos Maximiliano Correa, Pablo Leonel Etcheverry y Carlos Etcheverry.*

#### Track 5 – Enfoques y Entornos / Computación y sociedad

Moderadores: Mgter. Beatriz Castro Chans (UNNE), Lic. María Ferraro (UNNE)

La Escuela de Robótica de Misiones como modelo de implementación de metodologías disruptivas de enseñanza y aprendizaje. *Autores: Solange Schelske, Sebastian Rossi y Marcelo Benítez.*

Didáctica de la Programación en Educación Inicial: la co-construcción entre estudiantes y docentes. *Autores: Cecilia Haydée Exeni*

De la basura electrónica a juegos. *Autores: Claudio Javier Di Paolo.*

Identificación de sensores utilizados en robótica educativa mediante aprendizaje automático. *Autores: Javier Berger y Solange Schelske.*

## La TGS como base para el pensamiento computacional en la formación de profesionales informáticos

Flavia Sabrina Moreiro (FaCENA – UNNE) (flaviamoreiro@exa.unne.edu.ar)

Beatriz Castro Chans, (FaCENA – UNNE) (beatrizcc@exa.unne.edu.ar)

María L. Guastavino Mosna (FaCENA – UNNE) (maria.lorena.guastavino@comunidad.unne.edu.ar)

Guillermo Andrés Arduino (FaCENA – UNNE) gaarduino@exa.unne.edu.ar

### Resumen

El presente trabajo relata la experiencia de aplicación de una metodología para el análisis de las organizaciones como sistemas sociales, en el marco de la asignatura Sistemas y Organizaciones del primer año de la carrera Licenciatura en Sistemas de Información en la Facultad de Ciencias Exactas, Naturales y Agrimensura de la Universidad Nacional del Nordeste.

Durante el dictado de la asignatura se propone un trabajo práctico grupal, que consiste en la descripción y el análisis de una organización del sector software y servicios informáticos desde el enfoque de la Teoría General de Sistemas. Entendemos que la metodología aplicada por la asignatura contribuye al desarrollo del Pensamiento Computacional ya que la actividad implica desplegar una serie de destrezas particulares vinculadas a este concepto. A su vez, promueve el conocimiento de las competencias profesionales requeridas a los informáticos, muchas de las cuales se encuentran relacionadas con las destrezas requeridas por el Pensamiento Computacional.

Una encuesta anónima aplicada a los estudiantes al finalizar el cursado de la asignatura arroja como resultado que los estudiantes valoran positivamente al análisis de las organizaciones, destacando que les permite comprender su funcionamiento y los acerca al campo laboral donde se desempeñarán. Además, tanto en las producciones grupales de los estudiantes como en la presentación del coloquio final se pudo observar, una buena comprensión de los temas y la aplicación pertinente el análisis sistémico de las organizaciones.

**Palabras clave:** Sistemas - Informática - Organizaciones - Complejidad - Competencias

### Introducción:

El pensamiento computacional, según Wing *“implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, basándose en los conceptos fundamentales de la ciencia de la computación. El pensamiento computacional incluye una amplia variedad de herramientas mentales que reflejan la amplitud del campo de la computación... [además] representa una actitud y unas habilidades universales que todos los individuos, no sólo los científicos computacionales, deberían aprender y usar”* (2006, p. 33)

Sena (2010) señala que la acción se orienta a reducir al máximo las tareas repetitivas a través de la

programación de las máquinas. Para ello es necesario desarrollar el pensamiento computacional.

Según afirma Bordignon (2020, p.26), las actividades que promueven el pensamiento computacional fomentan en las personas el desarrollo de una serie de destrezas particulares, entre las que se incluyen: confianza al trabajar con la complejidad, persistencia al trabajar con problemas difíciles, tolerancia a la ambigüedad, capacidad para lidiar con problemas abiertos y cerrados, capacidad para comunicarse y trabajar con otros para lograr una meta en común.

Por su parte, el mismo autor citando a Brennan y Resnick (2012) recupera un enfoque alternativo del pensamiento computacional que abarca tres dimensiones: *conceptos computacionales* (que son aquellos que emplean los diseñadores en el trabajo de programación); *prácticas computacionales* (las que desarrollan a medida que programan) y *perspectivas computacionales*, que son las que los diseñadores construyen sobre el mundo que los rodea y sobre sí mismos.

Algunos de los problemas que se buscan resolver a través del pensamiento computacional, corresponde a emergentes de sistemas sociales (empresas, instituciones, asociaciones). Los sistemas sociales son realidades que se entienden a partir de observaciones. Son observables sólo a través de ciertas prácticas o ciertos funcionamientos que nos permiten inferir que son parte de una estructura compleja. Para Luhmann, las comunicaciones son los elementos fundamentales de los sistemas sociales. Siguiendo esta línea de pensamiento, la estructura del sistema es el resultado de la propia dinámica de las relaciones entre las comunicaciones que se producen en su interior, en cada momento. (Dallera, 2010)

Esta perspectiva se inscribe en la Teoría General de Sistemas cuyos objetivos, según Arnold (1998) son: a. Impulsar el desarrollo de una terminología general que permita describir las características, funciones y comportamientos sistémicos. b. Desarrollar un conjunto de leyes aplicables a todos estos comportamientos y, por último, c. Promover una formalización (matemática) de estas leyes.

El enfoque sociológico del constructivismo sistémico (CS), utilizando conceptos provenientes de teorías sistémicas caracterizadas por su énfasis de los fenómenos dinámicos, las totalidades abiertas al entorno, los procesos complejos y las interacciones fuertes, aporta una concepción de los sistemas sociales como *autorreferentes* y *autopoiéticos* compuestos por comunicaciones. (Foio, 2008)

En este sentido, entendemos que la asignatura Sistemas y Organizaciones, contribuye al desarrollo del pensamiento computacional en tanto propone una metodología para el análisis de las organizaciones como sistemas sociales integrados a un entorno complejo y contingente. Además, promueve el conocimiento y el desarrollo de habilidades en consonancia con las competencias sociales, políticas y actitudinales establecidas en el Libro Rojo del CONFEDI (2018), tales como: desempeñarse de manera efectiva en equipos de trabajo; comunicarse con efectividad; actuar con ética, responsabilidad profesional y compromiso social, considerando el impacto económico, social

y ambiental de su actividad en el contexto local y global; aprender en forma continua y autónoma; y, actuar con espíritu emprendedor.

## **Descripción de la Experiencia**

La experiencia que se describe fue desarrollada en la asignatura Sistemas y Organizaciones, la cual se dicta en el primer año, segundo cuatrimestre, de la carrera Licenciatura en Sistemas de Información en la Facultad de Ciencias Exactas, Naturales y Agrimensura de la Universidad Nacional del Nordeste. La misma cuenta con una matrícula promedio de 300 estudiantes distribuidos en 3 comisiones.

La perspectiva de la complejidad es transversal a todo el programa de la asignatura y el abordaje específico de la Teoría General de Sistemas (TGS) se encuentra en el segundo módulo del programa, asignándole unas 6 semanas a su desarrollo y se retoma en el coloquio final integrador.

Durante el dictado de la asignatura, se desarrollan clases teórico-prácticas y se realizan talleres donde se proponen distinto tipo de actividades tales como lectura comprensiva de los temas teóricos, análisis de caso empleando estrategias de tipo colaborativas y utilizando aplicaciones web, trabajo en equipo, entre otros.

Entre dichas actividades, se propone un trabajo práctico grupal, que consiste en la descripción y el análisis de una organización desde el paradigma de la complejidad. Este tiene como objetivos, por un lado, conocer organizaciones de la región a través de un proceso espiralado que articule los conceptos teóricos y que permita visualizar el caso desde el enfoque sistémico, y por otro lado, analizar el perfil del profesional informático requerido por el campo laboral regional, en función de la formación ofrecida por la LSI.

Los equipos están integrados por entre 4 y 6 estudiantes y se conforman de acuerdo a sus propios intereses. Para ello, se trabaja en el contexto de cátedra, aspectos tales como, la diferencia entre grupo y equipo, la elaboración de normas de funcionamiento, las cuales son revisadas y actualizadas por ellos mismos durante la cursada.

Las organizaciones que son analizadas se dedican al sector software y servicios informáticos / al sector TIC, o bien, el análisis se centra en áreas informáticas o TIC de organismos públicos o privados de otros sectores de actividad. Las mismas están localizadas en las Provincias de Corrientes y del Chaco. Las organizaciones son seleccionadas por el equipo docente aplicando criterios de pertinencia (vinculación con el campo de la informática) y variedad (que permita establecer comparaciones en términos de su complejidad, estructura organizativa, cultura organizacional, gestión de autoridad, entre otros aspectos).

Para realizar el trabajo práctico se parte de los contenidos teóricos abordados en las clases, que luego deben ser aplicados a los casos de estudio. También, se requiere la búsqueda de información en los sitios web y redes de las organizaciones y se programan entrevistas con los referentes de las mismas

para profundizar acerca de su funcionamiento. Las entrevistas se realizan en el horario de clases, con una hora de duración aproximadamente, siendo de manera presencial hasta antes de la pandemia y en modalidad virtual los últimos dos años, dado el contexto. Previamente, se prepara colaborativamente un cuestionario a partir de la identificación del tipo de información necesaria para la descripción del caso.

El cuestionario a realizar a los referentes de las organizaciones, se elabora a partir de una serie de consignas y con la tutoría del docente a cargo de los trabajos prácticos de la comisión. En primera instancia, cada grupo elabora las preguntas que considera necesita para recabar la información que no pudo ser obtenida de fuentes secundarias (página web, publicaciones digitales, etc.). En una instancia de clase (presencial o virtual) los grupos ponen en común las preguntas elaboradas y se realiza un proceso de organización y jerarquización de las preguntas para facilitar la exposición, revisión de la sintaxis, análisis del cuestionario para identificar si quedan aspectos sin relevar, etc.

Además, una vez consensuado el cuestionario, se solicita a los estudiantes se postulen voluntariamente para actuar como entrevistadores (usualmente son dos o tres). Luego, se establecen algunas pautas que deberán considerar al momento de realizar las preguntas (presentarse y establecer un trato amable, prestar atención durante toda la entrevista para no realizar preguntas que ya hayan sido respondidas, agradecer al finalizar, etc.). Cabe aclarar que durante la entrevista, tanto el equipo docente como el resto de los estudiantes pueden incorporar otras preguntas que surgen de la propia exposición y se vinculan con el tema o despiertan su interés.

El trabajo de cada grupo es plasmado en un informe escrito, el cuál es retomado al finalizar el cursado en un coloquio integrador, junto al análisis de los demás contenidos de la asignatura, como así también es insumo para los exámenes finales.

¿Qué se analiza en los casos de estudio? Entendiendo a las organizaciones como sistemas de información desde la TGS, se busca analizar las relaciones sistema-entorno y las que se producen al interior del sistema, su capacidad para autoorganizarse y producir sus propios elementos, la diferenciación sistémica (creación de subsistemas), los procesos de comunicación y los procesos de aprendizaje organizacional (retroalimentación positiva y negativa), entre otros aspectos vinculados a los sistemas complejos.

La complejidad implica selectividad forzosa; da cuenta de la necesidad que tiene todo sistema, en cada uno de sus actos existenciales, de excluir diversas posibilidades para poder realizar alguna. Porello, la estabilidad de los sistemas ha de ser necesariamente dinámica, abierta a una constante comparación y búsqueda de alternativas, lo que obliga al análisis sistémico a utilizar como método fundamental el análisis funcional.

En este marco, se recorre la historia de la organización desde sus inicios hasta la actualidad, identificando situaciones de cambio, las decisiones tomadas ante dificultades o demandas tanto internas como externas, el manejo de tecnologías, así como también el rol de los informáticos en estas situaciones



y las competencias profesionales que se les requiere.

La organización social es estudiada en base a los procesos propios de los sistemas complejos, a partir de lo cual, mediante operaciones de apertura y clausura, se relacionan con el medio y procesan información necesaria para las actividades de rutina, la adaptación a diversas situaciones y problemas, el funcionamiento y las articulaciones de los subsistemas internos y la transformación de sus propias estructuras.

Además, en la actualidad, el mundo del trabajo demanda a la universidad graduados que hayan adquirido competencias generales, es decir, que hayan cultivado capacidades sociales y de comunicación, que estén preparados para desempeñarse en equipos de desarrollo multidisciplinarios, en estructuras flexibles, que puedan identificar y resolver problemas a partir de la comprensión y el análisis crítico de los comportamientos organizacionales, todo ello en los nuevos escenarios de la sociedad de la información y la globalización de los mercados. En ese marco, se impulsa a los estudiantes a reconocer e interpelar a las problemáticas de la sociedad en el mundo actual y en especial en la Argentina y la región NEA, conocer diversos espacios laborales y procesos de transformación, desafíos, limitaciones y posibilidades del contexto actual.

Se busca aportar una perspectiva global de sistemas donde se pueda apreciar el valor estratégico de la información y la gestión del conocimiento en la sociedad actual. A su vez, que los estudiantes puedan adquirir destrezas para observar las relaciones entre el entorno social y las organizaciones, reconocer los procesos y mecanismos que acontecen en su interior. Los mecanismos de reducción de la complejidad, los códigos propios de los diferentes subsistemas, los esquemas binarios de selección y regulación de las relaciones al interior de cada subsistema y los intercambios entre ellos, así como también comprender la función del principio de retroalimentación, en el cual reposa el modo de comportamiento de los sistemas sociales intencionales.

## **Resultados**

Para conocer la visión de los estudiantes sobre el desarrollo de la asignatura, al finalizar el cursado, se aplica una encuesta de carácter anónimo. Dado el contexto, los dos últimos años fue realizada mediante el aula virtual de la asignatura. La misma contiene mayormente preguntas cerradas y, también, preguntas abiertas.

A continuación, se presentan los resultados cuantitativos de una selección de preguntas cerradas de la encuesta, referidas específicamente a la aplicación de la TGS al análisis de las organizaciones. La misma fue aplicada en el año 2021 y respondida por 97 estudiantes sobre un total de 293 cursantes (341 matriculados).

El perfil de los estudiantes que finalizaron el ciclo de cursada en la asignatura en el año 2021 que se observa en el gráfico 1, es de un 75% de varones y un 24% de mujeres, con un 1% de los estudiantes que prefirió no informar su condición de género. De dicha población, el rango etario más frecuente es

entre los 21 y 30 años, seguido por quienes son menores a 20 años.

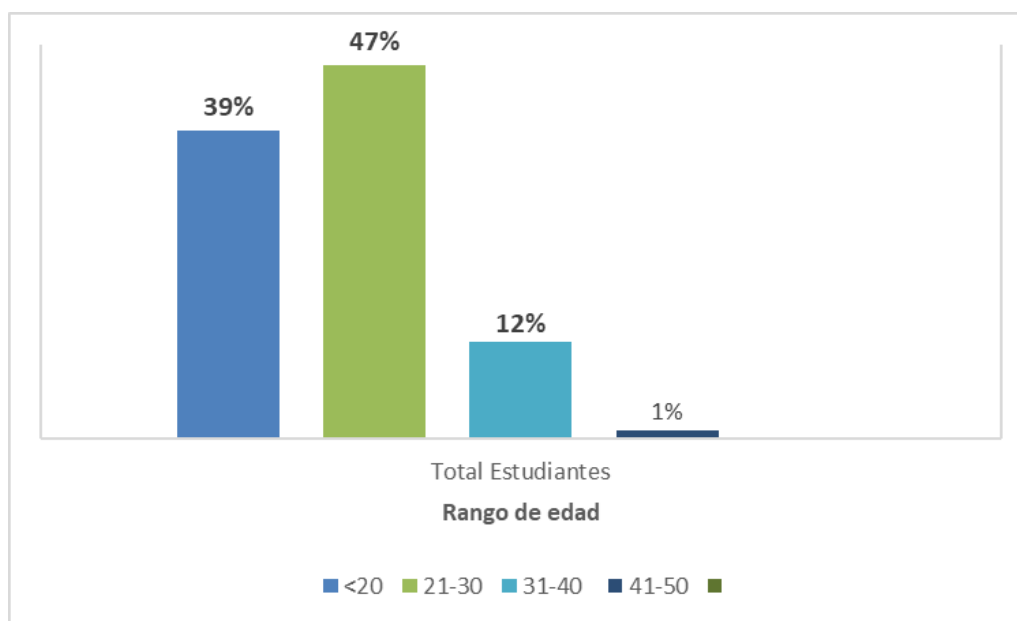


Gráfico 1. Porcentaje de estudiantes según rango etario

En cuanto a si los estudiantes poseían experiencia en haber cursado alguna otra carrera universitaria antes de la Licenciatura en Sistemas de Información, el 56% declaró haber cursado una carrera universitaria anterior a esta. Respecto a la pregunta sobre su experiencia laboral 41 % de los estudiantes manifestó trabajar. Ese porcentaje se distribuye entre quienes trabajan en el ámbito de la informática (11%) y quienes no lo hacen (30%). Cabe aclarar que 5 estudiantes optaron por no expresar su condición de empleo.

El gráfico 2, describe en qué medida la asignatura colaboró para conseguir determinados resultados de aprendizaje, según el siguiente detalle:

En cuanto al análisis de las organizaciones desde el enfoque sistémico, el 68% de los alumnos manifestaron que en gran medida pudieron alcanzar este resultado. Luego, el 65% expresó que los ayudó en gran medida a ampliar su visión del rol del informático. En relación a la competencia de trabajo en equipo, el 59% de los alumnos afirmó que en gran medida lo pudieron hacer. En cuanto así pudieron mejorar sus competencias de lectura, escritura y expresión oral, el 51% de los estudiantes indicó que en parte pudieron hacerlo y el 14% de ellos señalan que no pudieron mejorar estas competencias, siendo en este punto donde los estudiantes observaron mayor dificultad.

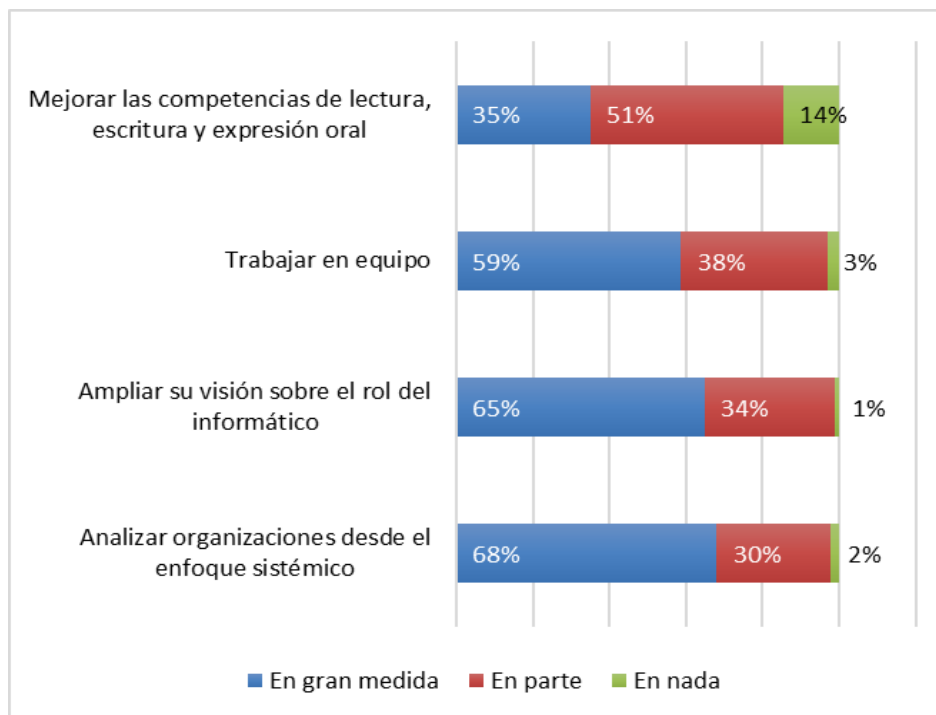


Gráfico 2. Resultados de análisis de casos desde la visión del estudiante

El gráfico 3 describe en qué medida las entrevistas realizadas a referentes de la industria del software fueron de importancia para su formación como informáticos y cómo colaboraron para conocer las características del campo laboral y su rol en las organizaciones.

En este sentido, 72% de los estudiantes expresaron que las entrevistas con los referentes de empresas ampliaron en gran medida su visión sobre el campo laboral y el rol profesional de los informáticos, mientras que 23% de los estudiantes indican que lo hicieron en parte.

Además, 68 %de los estudiantes manifestaron que las entrevistas con los referentes de empresas aportaron elementos importantes para tu formación profesional en gran medida. 30% de los estudiantes indican que aportaron en parte.

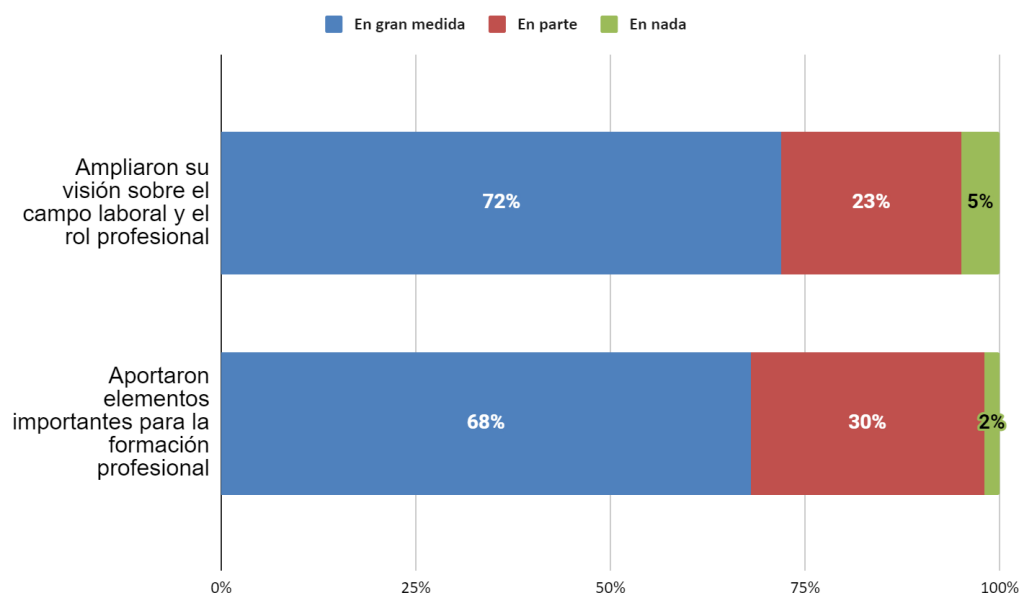


Gráfico 3. Aportes de las entrevistas a la formación profesional y a la visión del campo laboral.

Al distinguir por un lado los resultados de los estudiantes que trabajan en actividades informáticas, y, por otro lado, los resultados de los estudiantes que no trabajan y que no cursaron otra carrera antes de la Licenciatura en Sistemas de Información, se puede observar que la tendencia es la misma en relación a los resultados del total, ya que en ambos casos la mayoría coincide en que en gran medida las entrevistas realizadas a los referentes de las empresas ampliaron su visión sobre el campo laboral y el rol profesional y que aportaron elementos importantes para su formación profesional como informáticos.

Finalmente, el 79% de los encuestados expresaron que en gran medida la preparación y realización del coloquio final integrador les permitió comprender mejor los temas desarrollados durante el desarrollo de la asignatura y relacionarlos con el funcionamiento de las organizaciones, tal como se observa en el gráfico 4.

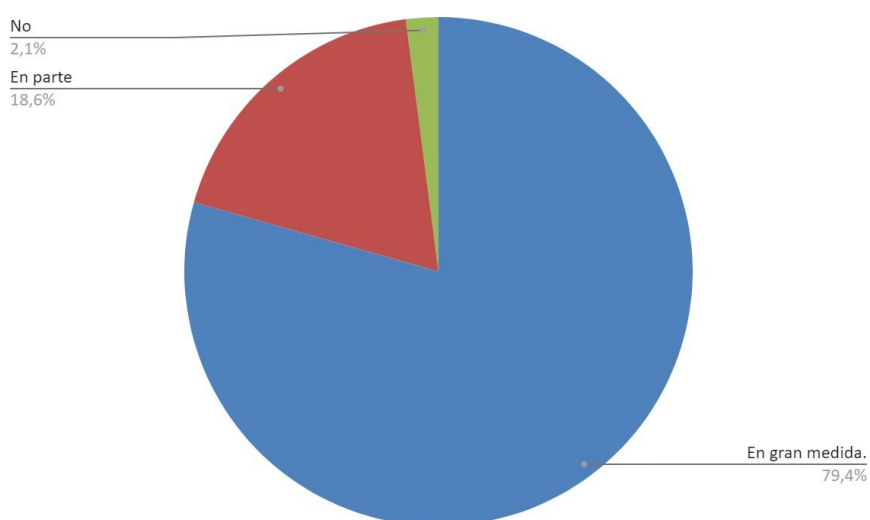


Gráfico 4. Contribución del coloquio a la comprensión de la dinámica organizacional

## Conclusiones

La sistematización de esta experiencia, permite constatar cómo la metodología aplicada por la asignatura, resulta útil a los estudiantes para conocer y desarrollar competencias profesionales, muchas de las cuales se encuentran en consonancia con las destrezas requeridas por el Pensamiento Computacional. Entre ellas se puede mencionar: apreciar la interacción entre teoría y práctica; trabajar de forma satisfactoria en equipo y con personas y en áreas de diversas disciplinas; poder analizar críticamente los procesos de solución de problemas y las soluciones en sí mismas; comunicar de manera eficaz ideas en forma escrita y oral, y valorar las presentaciones ajenas; elaborar resúmenes y realizar búsquedas bibliográficas; planificar las tareas y administrar el tiempo de forma eficiente; considerar la importancia del aprendizaje continuo; reconocer la responsabilidad de sus acciones; entre otras.

Estas destrezas y habilidades están fuertemente relacionadas con las competencias requeridas para los profesionales informáticos (CONFEDI, 2018).

Por su parte, los estudiantes valoran positivamente al análisis de las organizaciones realizado, destacando que les resulta un gran aporte para su formación, puesto que les permite comprender cómo funcionan y los acerca a la realidad laboral donde se desempeñarán profesionalmente.

Finalmente, además de los resultados obtenidos en las encuestas, se pudo observar tanto en las producciones grupales de los estudiantes como en la presentación del coloquio final en donde se integra y sintetiza el análisis sistémico de las organizaciones, una buena comprensión de los temas y la aplicación pertinente de los conceptos al caso de estudio.

## Referencias

- Adell, J. S., Llopis, M. A. N., Esteve, M. F. M., y Valdeolivas, N. M. G. (2019). El debate sobre el pensamiento computacional en educación. RIED. Revista Iberoamericana de Educación a Distancia, 22(1), pp. 171-186. doi: <http://dx.doi.org/10.5944/ried.22.1.22303>
- Arnold, M. (1998) Introducción a los Conceptos Básicos de la Teoría General de Sistemas. En: *Cinta de Moebio*. N°3. Abril de 1998. Facultad de Ciencias Sociales. Universidad de Chile.
- Bordignon, F., Iglesias A.A. (2020) *Introducción al pensamiento computacional*. - 1a edición para el profesor - Ciudad Autónoma de Buenos Aires: UNIPE; Editorial Universitaria; EDUCARS.E.
- CONFEDI (2018) *Propuesta de estándares de segunda generación para la acreditación de carreras de Ingeniería en la República Argentina "Libro Rojo de CONFEDI"*. Buenos Aires: Universidad Fasta Ediciones.
- Dallera, O. (2010) Sociología del sistema educativo (o crítica a la educación cínica). Buenos Aires: Biblos.
- Foio, M. (2008) *Las Organizaciones Sociales como Sistemas Complejos: El enfoque del constructivismo sistémico de Niklas Luhmann*. Apunte de cátedra de Sistemas y Organizaciones, FaCENA – UNNE.
- Serna, A. A. (2010). Línea del tiempo de las ciencias computacionales. Revista Digital *Lámpsakos*, No. 3, pp.86-94.
- Wing, J., M. (2006): Computational Thinking, Viewpoint Vol. 49, No. 3. Disponible en: <https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf>

# Educación Mediada en Primer Año de Ingeniería para la Formación de Competencias de la Industria 4.0 en el Contexto de la Enseñanza Remota de Emergencia

Artemisa Trigueros<sup>1</sup>, Mabel Compagnoni<sup>1</sup>, Larisa Toro<sup>1</sup>

<sup>1</sup> Universidad Nacional de La Matanza, Florencio Varela 1903,  
B1754 San Justo, Pcia. de Buenos Aires, Argentina  
{artemisa, mcompagnoni, lara, [@unlam.edu.ar](mailto:@unlam.edu.ar)}

## Resumen

*En la sociedad actual, las nuevas Tecnologías de la Información y de la Comunicación (TIC) han evolucionado vertiginosamente y el sistema educativo debe tener en cuenta dichos cambios e incorporarlos, con la perspectiva de favorecer los aprendizajes. Ante la necesidad de seguir impartiendo una educación de excelencia en el marco del aislamiento social, la Universidad implementó la enseñanza mediada por las TIC. Este artículo presenta información relativa a dicha experiencia, durante el primer cuatrimestre del año 2020, mediante ejemplos de actividades y resultados de una encuesta realizada a los alumnos de la cátedra Fundamentos de TICs correspondiente al primer año de las carreras de Ingeniería. La implementación de la educación mediada se realizó teniendo en cuenta los objetivos de la materia, para formar a los estudiantes en las competencias y en los conocimientos necesarios para incluirse en la Industria 4.0., que es el ámbito laboral y profesional del cuál van a ser parte.*

## Introducción

Las expectativas relacionadas con el uso y la aplicación de las TIC cambiarán las formas de interacción social, el trabajo y, por supuesto, la educación. Es por ello, que actualmente, la educación en todo el mundo debe enfrentar el desafío del uso de las tecnologías como herramienta pedagógica, con el fin de que los estudiantes desarrollen estrategias que le permitirán afrontar y satisfacer las necesidades de la sociedad futura.

El campo del diseño curricular muestra signos de avance, y tiene además elementos que permiten caracterizar su producción, pero también enfrenta retos [1]. Se podría considerar como uno de ellos a la implementación de la educación mediada en alumnos de primer año, ante la imposibilidad de la enseñanza presencial, que llevó a la necesidad de integración de las TIC en el modelo curricular. Es de gran ayuda ante este reto el hecho que, en general, los alumnos son como lo dice Prensky “nativos digitales”, que privilegian y dedican una gran parte de su

tiempo a actividades inherentemente relacionadas con nuevas tecnologías [2].

Otro de los retos a tener en cuenta se centra en los educandos, que también deben transitar procesos de aprendizaje en este cambio. Toda persona aprende haciendo, por lo que se deben estudiar e implementar herramientas pedagógicas que posibiliten potenciar y satisfacer los requerimientos de esta Cuarta Revolución Industrial, entendiendo la innovación y las tendencias; y a partir de ahí preparar a los estudiantes para desenvolverse satisfactoriamente en este mundo globalmente conectado y tecnológico.

Esta investigación específica la valoración que el alumnado asigna a las distintas herramientas pedagógicas provistas por las nuevas tecnologías, recoge sus opiniones respecto de las ventajas y desventajas de la cursada mediada basada en la Educación 4.0 y analiza la importancia de la interacción sincrónica entre alumnos y docentes durante las clases. Se concluye que una innovadora propuesta pedagógica sustentada con un adecuado uso de las plataformas tecnológicas y sostenida por un acompañamiento sincrónico del docente, permite una educación de calidad en el entorno virtual de enseñanza, jerarquizando, mediante las prácticas, los conceptos relacionados con los pilares de la Industria 4.0.

La materia Fundamentos de TICs (Tecnologías de la Información y la Comunicación), perteneciente al 1° Año de todas las Carreras de Ingeniería de la Universidad, representa uno de los primeros contactos académicos y formales, que tienen los alumnos con las TIC, cuya actividad profesional al graduarse, consistirá en desenvolverse en el marco de la “Cuarta Revolución Industrial” o Industria 4.0.

Paralelamente, la Educación 4.0 es una respuesta pedagógica a los requerimientos de competencias específicas de la Industria 4.0, colocando al estudiante en el centro del ecosistema de la educación superior [3].

La Educación 4.0 propone el aprendizaje activo, la cooperación como base del proceso de enseñanza, la comunicación e interacción entre docentes y alumnos, el aprendizaje competencial movilizando conocimientos para resolver problemas reales, utiliza el juego y los entornos de aprendizaje reales como motor de aprendizaje. Asimismo,

la evaluación es un proceso continuo de retroalimentación que promueve el progreso del estudiante.

En este entorno, las TIC están incluidas como herramientas educativas.

Particularmente, en este trabajo, las TIC cumplen una doble función, por un lado, están incluidas como herramientas de la Educación 4.0, por el otro son el objeto de estudio de la materia Fundamentos de TICs.

Estas tecnologías, incorporadas por la cátedra en el dictado curricular, tienen como objetivo mejorar y actualizar la transmisión de contenidos; facilitando el aprendizaje, la adquisición de nuevos conocimientos, promoviendo el pensamiento crítico y analítico, la creatividad, la originalidad y la inteligencia emocional, propuestas por la Educación 4.0. La implementación de tecnologías como realidad aumentada, realidad virtual, diseño de modelos, problemas de inteligencia artificial, impresiones 3D, Big Data, resolución de problemas reales en forma grupal, entre otros métodos implementados, colocará al alumno frente a situaciones reales de trabajo, formando en él las competencias requeridas por la Industria 4.0.

## Encuadre General

La asignatura Fundamentos de TICs, aparece planteada dentro del plan de estudios establecido como una materia del ciclo general de conocimientos básicos (CGCB) de las cinco carreras de Ingeniería de la Universidad: Ing. Civil, Electrónica, Industrial, Informática y Mecánica, formando parte de la currícula aprobada de Primer Año. Tiene como objetivo dotar al alumno de conocimientos informáticos y tecnológicos básicos (relacionados con temas que serán desarrollados en profundidad en años posteriores de la carrera), a fin de brindar un panorama general que será de gran utilidad en el perfil del ingeniero cuando egrese. En este sentido, recibirá una formación básica referida a organizaciones relacionadas con las tecnologías de la información y las comunicaciones, sistemas de información, software y hardware de los sistemas computacionales, teleinformática y comunicación de la información, inteligencia artificial y multimedia. Apunta a generar actitudes que lleven al alumno a tomar contacto con la realidad del mercado con las nuevas tecnologías y con las características de los principales grupos de productos existentes, que le permitirán desenvolverse en el medio que será su actividad futura y estar en condiciones de interpretar la mayoría de los conceptos que normalmente se emplean en los ambientes dedicados a las nuevas tecnologías de la información y la comunicación. La universidad, cuenta con un Campus Virtual, denominado MIeL[4] (Materias Interactivas en Línea), que ha permitido implementar la metodología de educación mediada

Asimismo, el Libro Rojo del CONFEDII [5] formula las siguientes competencias:

## COMPETENCIAS DE EGRESO

a) Genéricas: cada institución universitaria, en su marco institucional y del proyecto académico individual, determinará para sus carreras, la estrategia de desarrollo para asegurar competencias de egreso genéricas comunes a todas las carreras de ingeniería y necesarias para asegurar el perfil de egreso.

Estas competencias son:

### • Competencias tecnológicas

1. Identificar, formular y resolver problemas de ingeniería.
2. Concebir, diseñar y desarrollar proyectos de ingeniería.
3. Gestionar, planificar, ejecutar y controlar proyectos de ingeniería.
4. Utilizar de manera efectiva las técnicas y herramientas de aplicación en la ingeniería.
5. Contribuir a la generación de desarrollos tecnológicos y/o innovaciones tecnológicas.

### • Competencias sociales, políticas y actitudinales

6. Desempeñarse de manera efectiva en equipos de trabajo.
7. Comunicarse con efectividad.
8. Actuar con ética, responsabilidad profesional y compromiso social, considerando el impacto económico, social y ambiental de su actividad en el contexto local y global.
9. Aprender en forma continua y autónoma.
10. Actuar con espíritu emprendedor.

b) Específicas: El plan de estudios debe garantizar el desarrollo de las competencias específicas para las actividades reservadas definidas en la terminal y verificar el cumplimiento, además, de la formación en el proyecto académico de la carrera, de los alcances de título que defina la institución, con la profundidad y calidad propia de un título de ingeniero.

Precisamente, los ingenieros egresados de Universidad, se van a desenvolver profesionalmente dentro de la Industria 4.0, por lo tanto, sus competencias deben orientarse con los objetivos de la misma.

## Industria 4.0 y Educación 4.0

### A. Industria 4.0

En la historia de la Humanidad han existido hasta el momento tres revoluciones industriales que tuvieron como características las innovaciones técnicas.

En la Primera Revolución Industrial (Siglo XVIII) el motor a vapor de agua, en la Segunda Revolución Industrial

(principios del siglo XX), la producción en línea utilizando energía eléctrica. La Tercera Revolución Industrial se produjo de la mano de la digitalización de los procesos anteriormente analógicos (década de 1970).

La llamada Industria 4.0, es el resultado de la llamada 4<sup>ta</sup> Revolución Industrial, que tiene como objetivo, al igual que las anteriores, aumentar la productividad, pero en esta revolución, las TIC cobra un rol preponderante, mediante la utilización de las redes (Internet) que permitirán la comunicación entre hombres y máquinas en sistemas ciberfísicos, recolectando y analizando enormes cantidades de datos, que permitirán producir bienes tangibles e intangibles de mayor calidad a un costo más reducido.

La industria 4.0 se basa en las siguientes tecnologías:

- Big Data.
- Cyberseguridad.
- Fabricación por Adición.
- Fábricas digitalizadas que jerarquizan al factor humano.
- Internet de las Cosas.
- Realidad Aumentada.
- Robótica.
- Simulaciones.
- Software basado en la nube.

“La generación y adopción de estas tecnologías aumenta la productividad y competitividad de toda la economía, con el consecuente impacto positivo sobre la creación de empleos” [6].

Esta cuarta revolución, requiere que la flexibilidad sea la forma de actuar tanto de las personas como de empresas e industrias. Hay que adaptarse mucho más rápidamente de lo que se adaptaban las personas en otras épocas. Hoy, quienes se incorporan a las empresas, a las fábricas, trabajan en ellas durante un período, aprenden lo que consideran adecuado y cambian de trabajo, y requieren entonces para eso mucha flexibilidad. Ese factor va a ser imperativo para los próximos tiempos. [7].

En relación a las competencias a desarrollar y esta revolución: “La capacidad de trabajar en equipo según los métodos de colaboración propios de entornos laborales menos jerárquicos y estructurados, más tecnológicos y dinámicos se vuelve fundamental. Lo que hay que tener claro, sin embargo, es que la cuarta revolución industrial va de personas, no de máquinas”. [8].

## B. Educación 4.0

El proceso de enseñanza-aprendizaje se ha transformado, evolucionado con respecto a modelos anteriores, centrando su atención en modelos dirigidos al aprendizaje [9].

Dentro de esta evolución las Tecnologías de la Información y la Comunicaciones, se convierten en herramientas poderosas para la comunicación directa o

indirecta, entre docentes y alumnos, apoyada por medios auxiliares [10].

Dado que los estudiantes son “nativos digitales” [2], sus vidas se desarrollan, desde su nacimiento, rodeados por la tecnología, a la que consideran parte de su mismo cuerpo.

Asimismo, los estudiantes poseen “...una habilidad innata en el lenguaje y en el entorno digital. Las herramientas tecnológicas ocupan un lugar en sus vidas y dependen de ellas para todo tipo de cuestiones cotidianas como estudiar, relacionarse, mensajearse, comprar, informarse o divertirse” [11].

Las TIC sirven para motivar y estimular al estudiante a involucrarse totalmente en el proceso, interactuando con la realidad y observando los resultados de esta interacción a fin de desarrollar habilidades de pensamiento crítico y creativo y de esta manera, facilitar la comprensión de lo que se ha aprendido de manera integral y dinámica [12].

Es necesaria la búsqueda e implementación de innovaciones educativas basadas en las tecnologías que contemplem las nuevas actitudes y expectativas de los alumnos, así como sus nuevas competencias. Dichas estrategias y recursos educativos, deben promover respuestas educativas innovadoras que se encuentren alrededor de alguno de los ejes siguientes:

- Eje contextual: para hacer más flexibles los recursos funcionales (tiempo y espacio) de forma que se generen nuevos dispositivos organizativos en materia de actividades de enseñanza y aprendizaje.
- Eje curricular: con la intención de ajustar los requerimientos curriculares para incorporar nuevas herramientas o contenidos culturales que respete las nuevas aptitudes del alumnado.
- Eje centrado en el proceso: propuestas para acomodar mejor los procesos y las actividades de aprendizaje a los cambios en las prácticas cognitivas y de comunicación, incluyendo tanto la comunicación interpersonal como la gestión del conocimiento, para beneficiarse así de sus mayores competencias en el ámbito de las tecnologías. [13].

“El docente tiene que explorar un nuevo desafío que plantea la educación actual. La baja dosis atencional de los estudiantes nos obliga a reelaborar nuestras clases repensando formatos que despierten la curiosidad y el interés del alumnado por los contenidos que les presentamos” [14].

La Educación 4.0 propone la cooperación como base del proceso de enseñanza. Permite la interacción constante entre alumnos y profesores, centrándose en la comunicación como principal vehículo para el aprendizaje. Aborda el aprendizaje competencial movilizando conocimientos para resolver problemas y casos reales. Busca el aprendizaje activo que pone al alumno a regular su proceso a través del pensamiento estratégico. Usa el juego y la creación de entornos de aprendizaje reales como motor de aprendizaje. Entiende la evaluación como un proceso de retroalimentación constante que ayuda a mejorar y



progresar. Utiliza las TIC como herramientas de acceso, organización, creación, difusión de contenidos. Primero define los objetivos y criterios de evaluación, después selecciona los contenidos y diseña las actividades de aprendizaje y finalmente piensa que herramientas tecnológicas pueden facilitar este proceso.

De todo lo anterior podemos deducir que la nueva educación tendrá que convivir de forma equilibrada con las tecnologías orientadas a las Tecnologías de la información, la comunicación y las conexiones, la Robótica, la Inteligencia Artificial, los Sistemas Expertos, la Realidad Virtual, la Realidad Aumentada y Big Data, entre otras.

Como conjunción de los temas expuestos, la cátedra de Fundamentos de TICs, que abarca gran parte de los temas que conciernen a la Industria 4.0, investiga las herramientas utilizadas en esa industria para estudiar los temas de la misma, logrando el doble objetivo de conocer nuevos contenidos a partir de prácticas sobre y con los mismos.

## Estrategias Didácticas

Las estrategias didácticas de esta propuesta pedagógica se basan en la Educación 4.0 que es una respuesta a los requerimientos de la industria 4.0 y coloca al estudiante en el centro del ecosistema de la educación superior como protagonista del mismo.

La Educación 4.0 propone fomentar pensamiento crítico y analítico, la creatividad, la originalidad, la inteligencia emocional y la metacognición como base del aprendizaje significativo.

Las TIC están incluidas como herramientas pedagógicas y, al mismo tiempo, son el objeto de estudio de la materia Fundamentos de TICs, como por ejemplo: realidad aumentada, realidad virtual, diseño de modelos, problemas de inteligencia artificial, impresiones 3D, Big Data.

Las estrategias propuestas fueron:

- La resolución de problemas reales en forma grupal a través de medios tecnológicos, simuladores, aplicaciones móviles, Campus Virtual, creación de videos, etc. que coloquen al alumno frente a situaciones reales de trabajo, formando en él las competencias requeridas por la Industria 4.0.
- La utilización de entornos de aprendizaje reales, casos, juegos y desafíos para promover el aprendizaje de competencias por medio de resolución de problemas del ámbito ingenieril.
- Las estrategias de aprendizaje realizadas con base en metodologías activas.
- La cooperación como base del proceso de enseñanza, plasmada por medio de la comunicación e interacción entre docentes y alumnos.
- La generación, por parte del docente, de un contexto favorable al aprendizaje donde los estudiantes sean los protagonistas.

- La transferencia de conceptos siempre girando alrededor de actividades como problemas y ejercicios, favoreciendo la retroalimentación que promueve el progreso del alumno.
- El espacio de reflexión con un lugar fundamental.
- El contenido siempre referido a uno o más tipos de problemas definidos.
- El trabajo en equipo, para que los estudiantes resuelvan problemas y ejercicios.
- La inclusión de Rúbricas y/o Listas de Cotejo en cada Trabajo Práctico Obligatorio, explicitando al alumno y grupo de desarrollo, los contenidos y actitudes a desarrollar y los criterios de evaluación de cada contenido.
- Charlas, exposiciones y demostraciones con los profesionales del polo Tecnológico en temas como Industria Aditiva (Impresión 3D), Realidad Aumentada y Realidad Virtual.

## Unidades

La materia Fundamentos de TICs cuenta con 6 (seis) Unidades Temáticas. Cada una de ellas comprende un tema específico de las TIC.

El desarrollo de contenidos comienza presentando una introducción a la Teoría de Sistemas de Información, considerando que todos los conocimientos incluidos en el currículo de la materia son, por sí mismos, sistemas.

Las siguientes unidades presentan Sistemas de Representación de la Información y Códigos para su transmisión segura, introduciendo al estudiante en el mundo digital y binario, continuando con Circuitos Lógicos, ya que por medio de éstos se diseñan los sistemas de procesamiento de datos y control de procesos; luego Hardware, integrando las unidades anteriores, a continuación Redes de Comunicación para interconectar sistemas y por último, Software, necesario para que dichos sistemas se desempeñen en forma automática.

## Aplicaciones de Herramientas de la Educación 4.0 a las Unidades de Fundamentos de TICs para crear competencias requeridas por la Industria 4.0

Los temas curriculares de la materia Fundamentos de TICs, corresponden al Plan de Estudios 2009. En estos 11 años, la sociedad y la tecnología han realizado grandes cambios. Basta pensar en la difusión de los dispositivos móviles, que han cambiado el perfil del alumno, y de los docentes, al permitir tener en la mano, toda la información y los recursos comunicativos necesarios al instante. Sin embargo, los temas curriculares se mantienen fijos desde 2009.

Para lograr el objetivo de creación de competencias requeridas por la Industria 4.0 en los estudiantes de 1<sup>er</sup>. Año,

la cátedra de Fundamentos de TICs, incorporó casos y ejemplos de la Industria 4.0 y metodologías activas de la Educación 4.0, que pertenecen a los grandes temas curriculares de la materia.

Estos cambios se basan fundamentalmente en la figura de Trabajos Prácticos Obligatorios (TPO), donde se plasman distintas tecnologías y metodologías, y en charlas de transferencia profesional y exposiciones de Ingenieros y equipos de investigación pertenecientes al Polo Tecnológico de la Universidad.

Los TPO, tienen la característica imprescindible de ser Trabajos Prácticos Grupales, ya que se persiguen crear en los alumnos recién incorporados a la Universidad, las siguientes competencias sociales, políticas y actitudinales:

- Desempeñarse de manera efectiva en equipos de trabajo.
- Comunicarse con efectividad.
- Actuar con ética, responsabilidad y compromiso social, considerando el impacto económico, social y ambiental de su actividad en el contexto local y global.
- Aprender en forma continua y autónoma.
- Actuar con espíritu emprendedor.

Asimismo, se busca comenzar a incorporar las siguientes competencias tecnológicas:

- Identificar, formular y resolver problemas de ingeniería.
- Concebir, diseñar y desarrollar proyectos de ingeniería.
- Gestionar, planificar, ejecutar y controlar proyectos de ingeniería.
- Utilizar de manera efectiva las técnicas y herramientas de aplicación en la ingeniería.
- Contribuir a la generación de desarrollos tecnológicos y/o innovaciones tecnológicas.
- Promover el aprendizaje activo a través del pensamiento estratégico aplicando herramientas tecnológicas.

A continuación, se describen como ejemplo en Tabla 1, los TPO pertenecientes a cada unidad temática implementados durante el primer cuatrimestre del año 2020. Es de destacar que dichos TPO, pueden incluir otros temas y/o casos de ejercitación y/o otras tecnologías a medida que éstas van surgiendo o como inquietud de los alumnos.

**Tabla 1. TPO por cada Unidad**

	ACTIVIDAD REALIZADA
UNIDAD 1 TPO1	Resolución de casos similares a los reales en los que el alumnado debe aplicar los temas vistos en la unidad 1 referidos a Sistemas de Información para identificar, formular y resolver problemas de ingeniería.
UNIDAD 2 TPO2	Aplicación en un caso concreto, utilizando códigos BCD, códigos QR y realidad aumentada. (Aprendizaje creativo basado en problemas concretos).

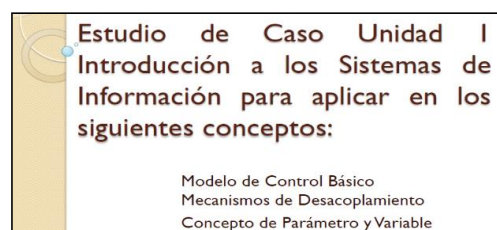
UNIDAD 3 TPO3	Utilización de software Logisim para simular la creación de circuitos lógicos, para la simplificación de funciones algebraicas.
UNIDAD 4 TPO4	Investigación sobre Periféricos. Armado de un informe de la investigación colaborativo grupal y publicación del mismo en un Foro público de acceso para los alumnos de cada comisión, con posterior retroalimentación por parte de sus pares. Opcional: este TPO incluye Transferencia Profesional sobre Impresoras 3D, consistente en una charla brindada por profesionales del Polo Tecnológico de la Universidad.
UNIDAD 5 TPO5	Confección de una investigación grupal referida a herramientas tecnológicas de la Web 2.0 sobre aplicaciones de IoT (Internet de las Cosas) y Ciberseguridad. Presentación de un informe de la investigación colaborativo grupal y publicación del mismo en un Foro público de acceso para los alumnos de cada comisión, con posterior retroalimentación por parte de sus pares. Opcional armado de un Video colaborativo grupal y publicación del mismo en una pizarra digital, con posterior retroalimentación por parte de sus pares que se introducía como comentario del postcard incorporado en la pizarra por cada grupo conteniendo el video colaborativo realizado.
UNIDAD 6 TPO6	Confección de una investigación grupal sobre casos de utilización de Inteligencia Artificial, Robótica, Big Data, Sistemas Expertos y Software Basado en la Nube por las empresas. Presentación de un informe de la investigación colaborativo grupal y publicación del mismo en un Foro público de acceso para los alumnos de cada comisión, con posterior retroalimentación por parte de sus pares.

Asimismo, se destaca que los alumnos aplicaron competencias adquiridas en los primeros TPO a los TPO siguientes.

## Ejemplos de trabajos prácticos obligatorios y testimonios de transferencia profesional

### Ejemplo del TPO 1: Parte B:

**Consigna del Proyecto:** Los alumnos se reunirán en grupos de 4 o 5 integrantes y dispondrán de 30 minutos para debatir en forma colaborativa acerca de las posibles soluciones para el estudio de caso planteado a través de las situaciones enunciadas a continuación, aplicando los conceptos analizados en la unidad 1 que consideren útiles para que el Ingeniero Oportuno pueda cumplir en forma óptima su tarea. Luego cada grupo presentará por escrito lo producido. La Figura 1 muestra los temas comprendidos en el TPO y las Figura 2 muestra el planteo inicial del caso de estudio.



**Figura 1. Temas abarcados en el TPO1.**

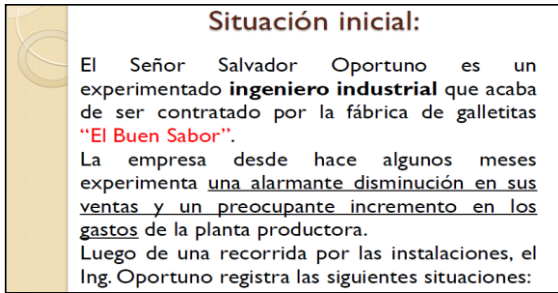


Figura 2. Planteo inicial del caso en TPO1.

La Figura 3, muestra parte del Foro de la Plataforma MIE<sup>2</sup> (Campus Virtual de Universidad, donde los alumnos intercambian ideas sobre la resolución del Trabajo Práctico Obligatorio planteado. Y la Figura 4 muestra el Foro del Curso 03 sobre el TPO y permite observar intercambio de ideas entre los alumnos.

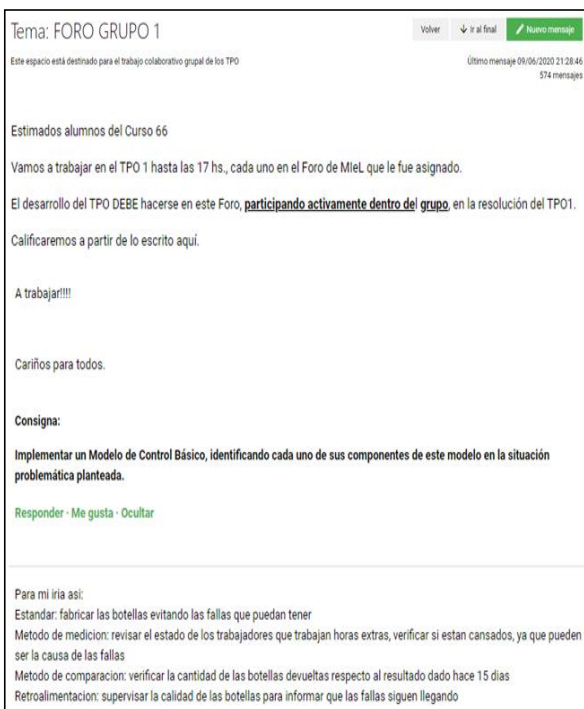


Figura 3: Foro del TPO1.



Figura 4: Parte del Foro de Curso 03. TPO1.

La Figura 5, muestra los enlaces a un foro del TPO4, que se comparten en trabajo colaborativo entre todos los alumnos del curso.

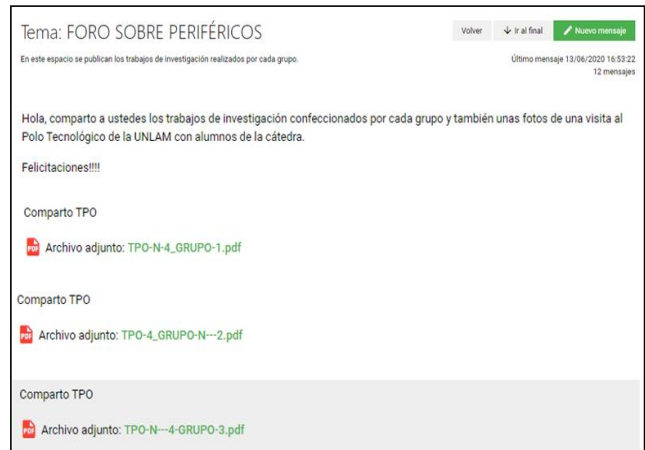


Figura 5: Foro del TPO 4.

Como puede observarse en la Figura 6, que contiene consignas del TOP2, el alumno debe escanear los códigos QR y observar aplicaciones de Realidad Aumentada aplicadas a la industria.



Figura 6: Consignas del TPO2.

Cada TPO incluye las rúbricas correspondientes, para informar y orientar a los alumnos sobre los niveles de logro y requisitos para alcanzarlos. La Figura 7 presenta las rúbricas correspondientes al TPO2.

FUNDAMENTOS DE TIC'S				
Herramienta de retroalimentación para el alumno: RÚBRICA del TPO 2				
Criterios	Niveles de logro			
	1	2	3	4
Utilización del Código QR e identificación de la información asociada	Dificultad en la utilización del Código QR y en la identificación de la información asociada al mismo	Identifica el Código QR pero presenta dificultad en la identificación de la información asociada al mismo	Identifica el Código QR pero sólo identifica la información asociada al mismo en forma parcial	Identifica el Código QR e identifica la información asociada al mismo en forma satisfactoria
Identificar y utilizar Códigos BCD	Dificultad en la identificación y utilización de los códigos BCD	Identifica los códigos BCD pero presenta dificultad en la utilización de los mismos	Identifica los códigos BCD pero los utiliza en forma parcializada	Identifica los códigos BCD y los utiliza en forma satisfactoria
Reconocer usos de los Códigos BCD	Dificultad en reconocer la utilización de los códigos BCD	Reconoce la utilización de los códigos BCD pero en forma escasa	Reconoce la utilización de los códigos BCD en forma parcializada	Reconoce la utilización de los códigos BCD en forma satisfactoria

Figura 7: Rúbricas para TPO2

Ante la nueva perspectiva de confeccionar un diseño tecno-pedagógico de enseñanza 4.0 mediada, como es el caso del diseño de ambientes y situaciones de aprendizaje en las plataformas MIEl y Teams<sup>3</sup> brindadas por la Universidad, se consideraron la naturaleza y el tipo de actividades que los alumnos realizarían de forma no sólo individual sino colectiva; las características de las herramientas que se pusieron a disposición del alumnado; los usos previstos de dichas herramientas; y las reglas y prácticas educativo-culturales en las que se insertarían las actividades propuestas.

Desde el punto de vista de los recursos tecnológicos que se pudieron incorporar en el diseño de entornos virtuales de enseñanza y aprendizaje, se destacaron aquellos que brindaron la posibilidad de integrar varias prestaciones comunicativas y colaborativas que facilitaron la comunicación sincrónica y asincrónica entre compañeros y profesores; tal es el caso de los chats, foros, mensajería, videoconferencias, entre otros. No obstante, es importante señalar que por sí mismos, y como meros recursos tecnológicos, no resolvieron el problema de la construcción conjunta del conocimiento, dado que lo relevante resulta ser el tipo de prácticas educativas y comunicativas que se propiciaron entre los participantes, es decir, los usos que se

<sup>3</sup> **Microsoft Teams** es una plataforma unificada que permite la colaboración entre personas de un mismo equipo, compartiendo

hicieron de dichas herramientas. Así pues, uno de los primeros aspectos relevantes en el diseño de entornos digitales tiene que ver, justamente, con la selección de uso que se les dará a las plataformas tecnológicas que dan soporte a la educación en un entorno virtual [15].

## Resultados

Comparando la cantidad de alumnos que Aprobaron y Cursaron la materia en los 1° Cuatrimestres de los años 2018, 2019 y 2020, podemos concluir que la implementación de la educación mediada en 2020 muestra mayor porcentaje de aprobación y cursado, que en los años anteriores, así como una disminución de Ausentes y Reprobados.

Se comparan los Primeros Cuatrimestres, ya que las condiciones de los alumnos son similares. Para la mayoría de ellos, es su primer cuatrimestre como alumno universitario.

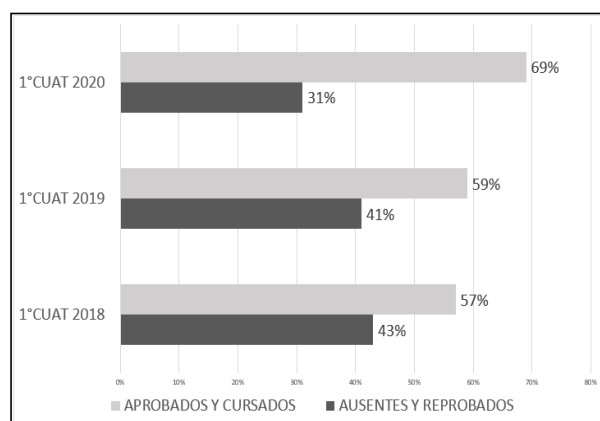


Fig. 8: Comparación de Porcentajes de Aprobados y Cursado con Ausentes y Reprobados en 1° Cuatrimestre 2018, 2019 y 2020.

El gráfico de barras de la Figura 8 muestra un 10% de crecimiento de Aprobados y Cursados y un 10% de disminución de Ausentes y Reprobados, con respecto a 2019. Es de destacar que la metodología de TPO, fue implementada a partir del 1° Cuatrimestre 2019. Ya en ese año, los porcentajes mejoraron, con respecto a 2018.

Durante la cursada fue realizada una Encuesta de Cátedra, donde los estudiantes valoraron la nueva metodología en forma muy positiva. La Figura 9 muestra los resultados de la encuesta por medio de un gráfico de sectores. La pregunta realizada fue si “Los TPO mejoran la comprensión de los temas.”

recursos y cuya función principal es la comunicación constante entre los miembros del equipo.



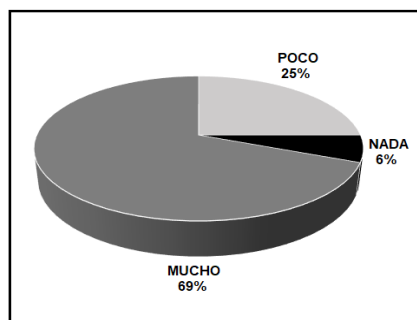


Figura 9: Consulta a los alumnos sobre los TPO.

Como puede observarse en la Figura 9 el 69% de los estudiantes respondió positivamente.

Además, para completar esta investigación se implementó una encuesta no obligatoria y anónima, dirigida a la totalidad de los alumnos de los siete cursos de primer año de la carrera de Ingeniería de la Universidad en que se dictó la materia Fundamentos de TICs durante el primer cuatrimestre del año 2020. De los cursos consultados en las encuestas, tres de ellos estaban conformados por alumnos ingresantes y otros cuatro cursos más constituidos por alumnos que recursaban la materia. Siendo la cantidad de encuestas respondidas por un total de 241 alumnos (73% de los alumnos que completaron el cursado de la materia). Se realizó una valoración cuantitativa y cualitativa de las respuestas obtenidas como muestran los resultados.

A continuación, se solicitó al alumno que asignara un puntaje a cada una de las herramientas tecnológicas implementadas en cursado virtual en Fundamentos de TICs y en otras materias, concediendo el valor 5 (mayor puntaje) a la herramienta que les ha brindado más ayuda y el valor 1 (menor puntaje) a la que les ha brindado menos ayuda. Los resultados obtenidos se pueden observar en la Figura 10 donde se representa el puntaje asignado por los alumnos a las herramientas utilizadas en las clases virtuales. Cabe destacar que estos porcentajes, representan la cantidad de alumnos que le asignaron la mayor valoración a cada una de las herramientas.

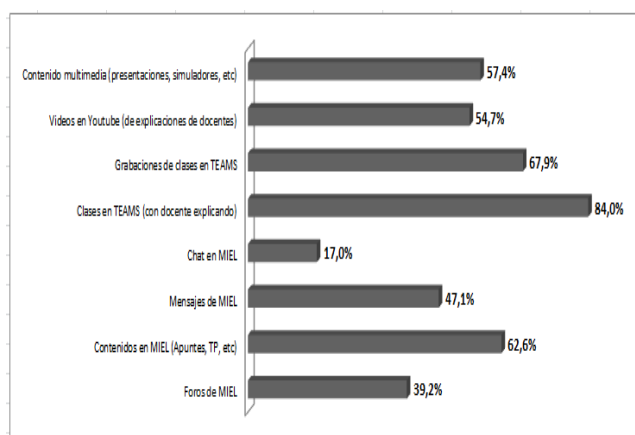


Figura 10: Puntaje asignado por los alumnos a las herramientas utilizadas en las clases virtuales. Se grafica el porcentaje obtenido por cada herramienta con la MAYOR puntuación.

Se observa que la herramienta que representó la mayor ayuda fueron las clases por videoconferencia en la plataforma Teams con el profesor explicando en vivo de manera sincrónica (84%), las grabaciones de las clases llevadas a cabo por videoconferencia (67,9%), los contenidos teóricos accedidos desde la plataforma MIEL (62,6%), los contenidos multimedia tales como presentaciones y simuladores (57,4%), videos en YouTube con explicaciones de los docentes de la cátedra (54,7%), mensajes en la plataforma MIEL (47,1%), foros en el entorno MIEL (39,2%) y Chats en la plataforma MIEL (17%).

En la siguiente pregunta se consultó a los alumnos qué servicios utilizaron dentro de la plataforma MIEL, la Figura 11 muestra los porcentajes de uso de cada uno de los servicios.

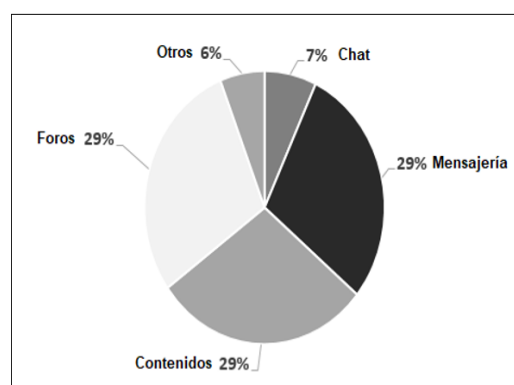


Figura 11: Servicios de MIEL utilizados por el alumno.

Se puede observar que los servicios más utilizados fueron foros, mensajería y contenidos: los tres servicios en la misma proporción.

Luego se preguntó qué metodología de cursado de las clases virtuales preferían, pudiendo optar entre las clases on line con el Profesor (clases sincrónicas) explicando como si el docente estuviera en forma presencial o bajar el material (guías de estudio, videos, presentaciones, material multimedia, etc.) y estudiar en forma autónoma (clases asincrónicas), La Figura 12 representa la preferencia de los alumnos por las clases sincrónicas con el docente explicando por videoconferencia (90%).

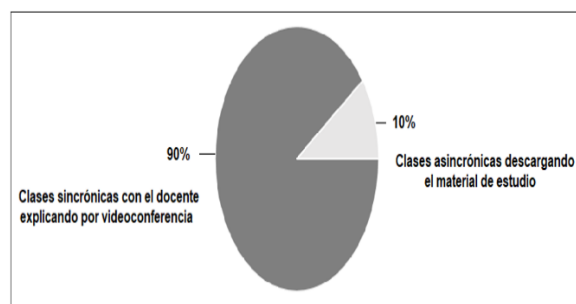
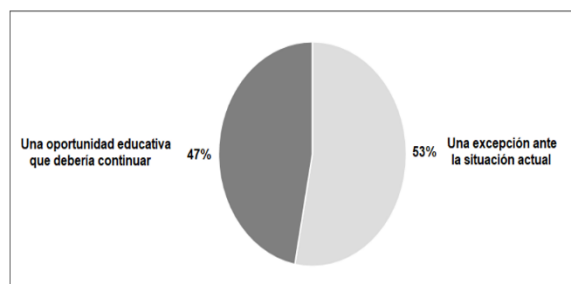


Figura 12: Preferencia del alumno respecto de la metodología de dictado de las clases virtuales.

Posteriormente se consultó al alumnado acerca de cómo consideraban a la modalidad virtual de cursado, como se grafica en la Figura 13.



**Figura 13: Consideración del alumno sobre la modalidad virtual.**

El gráfico permite observar que ambas opciones (“una oportunidad educativa que debería continuar” y “una excepción ante la situación actual”) recibieron valoraciones similares.

La encuesta realizada brinda como información que los alumnos mayoritariamente contaron con recursos tecnológicos que permitieron el cursado virtual de sus clases, pero lamentablemente revelan que frecuentemente experimentaron dificultades de conectividad en las zonas de residencia, lo que refleja una deficiente red de comunicación para sostener una educación virtual ante la imposibilidad de la presencialidad.

Entre los principales puntos positivos de la modalidad virtual se destaca que, al poder cursar desde sus casas, los alumnos hacen un mejor aprovechamiento del tiempo, esto se debe a que el aprendizaje ubicuo se relaciona también con una nueva forma de lectura, en la que se aprovechan los tiempos muertos de traslados o esperas, y se lee en espacios de tiempo denominados “burbujas de ocio” [16]. Otro aspecto señalado por el alumnado como positivo de la modalidad virtual enfatiza la disponibilidad y facilidad de acceso a los materiales didácticos (teóricos, multimedia y grabación de las clases por Teams). Esto se halla en concordancia con el aprendizaje ubicuo: “la posibilidad de acceder a la información en cualquier lugar o cualquier momento, la interacción con pares y expertos eruditos genera oportunidades estructuradas de aprendizaje desde una variedad de fuentes” [17].

La incorporación de nuevas herramientas tecnológicas puede aportar distintas modalidades y dinámicas para lograr nuevos conocimientos. Dado que “...los medios son extensiones de las capacidades del hombre...” y, como docentes, debemos promover acciones para extender esas capacidades [18]. Teniendo en cuenta que “...a medida que se utiliza frecuentemente una tecnología se la naturaliza...” [19].

El alumnado también resalta en sus respuestas abiertas, que los trabajos prácticos grupales y la claridad de los apuntes brindados, fueron de gran ayuda para la comprensión de los temas. Esto evidencia que la cátedra al elaborar la planificación pedagógica tuvo especial consideración que, en la modalidad de enseñanza mediada, es de vital importancia la gestión (diseño, elaboración, uso y evaluación) de los materiales didácticos ya que “...son uno de los ejes vertebradores de gran parte de las acciones de enseñanza desarrolladas en cualquiera de los niveles y

modalidades de educación...” [20]. Cabe destacar también, el consenso de los educandos al resaltar el compromiso y la buena predisposición de los docentes de la cátedra para explicar los temas y responder sus consultas. Demostrando que en este proceso de enseñanza-aprendizaje en un entorno virtual, el docente se desempeña como mediador y facilitador del trayecto formativo: “... el docente deberá transformarse en un guía o facilitador de los aprendizajes de sus alumnos...” [21].

## Conclusiones

La metodología de Educación 4.0 mediada implementada, fue recibida y ejecutada en forma muy satisfactoria por los estudiantes y docentes. Los resultados muestran que desde que se implementó los niveles de satisfacción fueron mejores que los de años anteriores. El proyecto fue un éxito desde el punto de vista de la cátedra y de los estudiantes.

El presente trabajo se orienta a promover una innovación de los docentes mediante la aplicación de la educación mediada, a fin de que puedan cambiar sus formas de enseñanza en las diversas áreas curriculares, que los estudiantes se sientan más interesados en sus aprendizajes, haciendo uso de nuevas estrategias y medios que permitan obtener un aprendizaje más significativo. Se orienta a producir cambios de actitud en los docentes en función de las necesidades de los estudiantes y del avance científico tecnológico, generando procesos actualizados e innovadores que contribuyan al logro de competencias, valores y actitudes en sus estudiantes que le permitirán una inserción social y laboral futura exitosa.

El modelo pedagógico que se implementó en el dictado de la materia, consolidó que es posible generar una propuesta conjunta de experiencias sincrónicas, llevadas a cabo a través de videoconferencias en la plataforma Teams, y otras experiencias asincrónicas a través de la plataforma MIEl, en el sentido de que por su propia complementariedad se potencian mutuamente, creando así un entorno abierto con múltiples intercambios comunicativos entre profesores y alumnos que promovió una mayor profundización y apropiación del conocimiento. Se concluye que la modalidad virtual puede brindar una trayectoria exitosa en el cursado de las materias de primer año en las carreras de Ingeniería, produciendo una educación mediada de calidad.

A través del uso pedagógico de las TIC, han surgido nuevos planteamientos pedagógicos que resaltan la prioridad del aprendizaje sobre la enseñanza; el papel clave del alumnado y la consiguiente necesidad de acudir a metodologías en las que el estudiante, sobre todo el universitario, tome las riendas de su aprendizaje con la dirección, la orientación y el apoyo de su profesor. Se privilegia el proceso formativo basado en una evaluación continua y en la continua propuesta de tareas. [22]

Mediante las nuevas herramientas tecnológicas, se prioriza el papel protagónico del alumnado en el proceso de aprendizaje, de todos los niveles, pero, en particular, del universitario, en la construcción del propio aprendizaje como fruto de sus lecturas, de su reflexión y del apoyo de compañeros y profesores, y continúa con la relación entre enseñanza y aprendizaje y en particular en el aprendizaje de competencias.

Del mismo modo cabe señalar la importancia, junto al aprendizaje autónomo, del aprendizaje colaborativo, que añade a los logros académicos repercusiones educativas de notable relieve, como el compañerismo, la solidaridad y la capacitación para el trabajo en equipo.

Las herramientas tecnológicas de la Educación 4.0 tienden a favorecer un aprendizaje más complejo e integrador basado en las asociaciones, comparaciones y un pensamiento crítico y reflexivo. Se promueve el desarrollo de la capacidad de análisis, lo que permite a los alumnos, conocer más profundamente y construir nuevos conocimientos a partir de otros previos. Esta modalidad de aprendizaje está relacionada con varias competencias (pensamiento crítico, resolución de situaciones de estudio, organización y argumentación de sus conclusiones, habilidad en la percepción de los detalles y de relaciones novedosas entre elementos propios de la realidad objeto de estudio, identificación de los elementos constitutivos del contexto estudiado, organización de la información significativa según criterios preestablecidos adecuados a un propósito). Y, por último, tener en claro los objetivos del tema analizado, que ayudarán a establecer criterios para seleccionar la información relevante y organizarla en la construcción de la síntesis.

La metodología de Educación 4.0 constituye un instrumento que permite profundizar respecto a las características del proceso de aprendizaje que los alumnos van desarrollando, lo que le posibilita al docente adquirir información valiosa para reorientar el proceso de enseñanza y colaborar en el logro de aprendizajes.

## Trabajo Futuro

La cátedra se propone seguir avanzando en la aplicación de nuevas herramientas y perfeccionando los casos planteados en los TPO.

Asimismo, se propone realizar nuevos cursos de capacitación docente de aplicación de metodologías de Educación 4.0.

## Referencias

- [1] Educación digital y cultura de la innovación - Colección Innovación Práctica Perspectivas de futuro - Fundación Cotec para la Innovación Tecnológica (2014)
- [2] Prensky, M., «Digital natives, digital immigrants», On the Horizon, núm 9, pp.1-6. (2001)

- [3] <https://ilab.net/la-universidad-del-futuro-y-la-educacion-4-0/>
- [4] [MIEl. Materias Interactivas en Líneas](http://miel.unlam.edu.ar)
- [5] Libro Rojo de CONFEDI. [https://confedi.org.ar/download/documentos\\_confedi/LIBRO-ROJO-DE-CONFEDI-Estandares-de-Segunda-Generacion-para-Ingenieria-2018-VFPublicada.pdf](https://confedi.org.ar/download/documentos_confedi/LIBRO-ROJO-DE-CONFEDI-Estandares-de-Segunda-Generacion-para-Ingenieria-2018-VFPublicada.pdf)
- [6] G. Nieponice. <https://www.infopymes.info/2018/08/que-es-la-industria-4-0-y-cuanto-falta-para-que-llegue-a-la-argentina/>
- [7] L. De Bernárdez. <https://www.infobae.com/def/ desarrollo/2019/01/08/la-industria-4-0-una-revolucion-en-las-fabricas/>
- [8] <https://www.lavanguardia.com/tecnologia/20190307/468930357/12/industria-40-empleo-formacion-profesional-profesionales-mano-de-obra-espana.html>
- [9] Edel, R. *El Concepto de Enseñanza-Aprendizaje*. Revista Electrónica Red Científica. Ciencia, Tecnología y Pensamiento. 2004
- [10] Betancur, D., Moreno, J. & Ovalle, D. A. Modelo para la Recomendación y Recuperación de Objetos de Aprendizaje en Entornos Virtuales de Enseñanza/ Aprendizaje. Revista Avances en Sistemas e Informática. 2009.
- [11] García, M. R. Interacción y Comunicación en Entornos Educativos: Reflexiones Teóricas, Conceptuales y Metodológicas. Revista de Asociación Nacional de Programas de Posgrado en Comunicación (E-Compós). Vol. 8. Disponible en <https://doi.org/10.30962/ec.143>. 2007.
- [12] Esteve, F.M. & Gisbert, M. (2011) El Nuevo Paradigma de Aprendizaje y las Nuevas Tecnologías. Revista de Docencia Universitaria. REDU. Vol.9 (3). Pag.s 55 – 73.
- [13] Francese Pedro: "Aprender en el nuevo milenio. Un desafío a nuestra visión de las tecnologías y la enseñanza"
- [14] Lazzatti, Pablo. "Jugarle a la cabeza: una apuesta al aprendizaje lúdico y las neurociencias en la enseñanza del Derecho." Anales de la Facultad de Ciencias Jurídicas y Sociales (2017).
- [15] Esteve, F.M. & Gisbert, M. (2011) El Nuevo Paradigma de Aprendizaje y las Nuevas Tecnologías. Revista de Docencia Universitaria. REDU. Vol.9 (3). Pag.s 55 – 73.
- [16] Igarza, R.: Burbujas de ocio. Nuevas formas de consumo cultural. Buenos Aires. Ediciones La Crujía (2009)
- [17] Burbules, N.: El aprendizaje ubicuo y el futuro de la enseñanza. Rencontres on Education. Vol. 13, pp. 3–14 (2012)
- [18] McLuhan, M.: Comprender los medios de comunicación. Las extensiones del ser humano. Barcelona. Ed. Paidós (1994)
- [19] Ong, Walter.: Oralidad y Escritura. Fondo de cultura económica. Capítulo 4 (1994)
- [20] Área Moreira, M.: Manual electrónico: Introducción a la Tecnología educativa, p24, Universidad de La Laguna (España) (2009)
- [21] Marín Díaz, V., & Romero López, M. A.: La formación docente universitaria a través de las TICs. Pixel. Bit. Revista de Medios y Educación. (2009)
- [22] Pérez Justé, R. (Coord.) (2012). El portafolio. Aprendizaje, competencias y evaluación. Universidad de Educación a Distancia (UNED), Madrid, España. Capítulo 4

## **Gestión del tiempo en actividades Remotas de estudiantes ingresantes de la Carrera Lic. en Sistemas de la FaCENA – UNNE**

Del Grosso, Ma. Ornella; Mosquera, María del Mar; Fernández, Mirta.

Gabinete Psicopedagógico. Facultad de Ciencias Exactas y Naturales y Agrimensura. 9 de Julio 1449.  
Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura. 9 de Julio 1449  
3400. Corrientes– Argentina  
Universidad Nacional del Nordeste.

[ornelladelgrosso77@gmail.com](mailto:ornelladelgrosso77@gmail.com); [mariadelmarmosquera15@gmail.com](mailto:mariadelmarmosquera15@gmail.com); [mirtagf@hotmail.com](mailto:mirtagf@hotmail.com)

### **Resumen**

Los espacios virtuales de aprendizaje favorecen aspectos diferentes a los espacios de clases presenciales. Al mismo tiempo, surgen otras variables: saber manejar y planificar el tiempo como aspectos estratégicos en el rendimiento académico. Este trabajo, pone en relevancia la gestión del tiempo de estudiantes ingresantes que realizan actividades remotas complementarias a la presencialidad; al mismo tiempo reconoce la importancia de las habilidades blandas y la capacidad de autorregulación como lineamientos que favorecen al desempeño académico en tal contexto. En consecuencia, el análisis busca conocer la gestión del tiempo en actividades remotas complementarias al ingreso y permanencia en la Universidad. Para ello, se trabajó con estudiantes de la carrera de Licenciatura en Sistemas de la FaCENA – UNNE, a fin de obtener un diagnóstico que facilite la descripción e interpretación de la percepción de los estudiantes respecto a cómo transitan estos nuevos escenarios combinados desde la presencialidad y la virtualidad como complemento voluntario. Se desarrollaron talleres destinados a fomentar la apropiación de métodos y estrategias aplicables en tal sentido, y basados en ellos este trabajo presenta algunas conclusiones preliminares.

### **Palabras Claves:**

Gestión del Tiempo. Actividad Remota. Estudiantes Ingresantes.

### **1. Introducción**

Desde el inicio del ciclo académico 2022, la FaCENA registró un 40.8% de estudiantes ingresantes a la Carrera de Licenciatura en Sistemas, del total que presentó la Unidad Académica. Ante este porcentaje cabría desarrollar varios interrogantes: preguntarse sobre la gestión del tiempo de dichos estudiantes, tras combinar la presencialidad y actividades de modalidad remota; además, si ser estudiante de primer año repercute en dicha gestión del tiempo, si se parte del hecho que se requiere de la



puesta en juego de habilidades blandas, como ser la adaptabilidad, para la regulación y planificación del estudio en entornos virtuales de aprendizaje.

Asimismo, corresponde destacar que existe una gran demanda laboral de recursos humanos con perfiles calificados para el sector de Software y Servicios Informáticos (SSI). Debido a que las Tecnologías de la Información y Comunicación (TIC) contribuyen significativamente al crecimiento del PBI en la mayoría de los países. En efecto, las habilidades blandas (*soft skills*) son atributos personales que el mercado valorará, ya que los profesionales deben manifestar competencias que aseguren un trabajo eficiente. Para ello, es fundamental complementar las habilidades duras (*hard skills*), adquiridas en el proceso de enseñanza-aprendizaje, con el desarrollo de las anteriores.

Actualmente, un gran porcentaje de jóvenes hacen uso de dispositivos de tecnología informática. Sin embargo, se podría suponer que la problemática se manifiesta al momento de realizar actividades formales, o sostenerse desde un aprendizaje autorregulado, demostrando tener cierta dificultad. En este sentido, en el caso de estudiantes que realicen estudios o labores prácticos virtuales, la gestión del tiempo y el despliegue de las habilidades blandas se constituye en un verdadero reto. Deben aprender a regular el tiempo disponible para la realización de tareas y evadir las situaciones emergentes de distracción que estos dispositivos ofrecen. Del mismo modo, deben aprender a desarrollar, y equilibrar, habilidades blandas y duras a fin de alcanzar eficiencia y eficacia a futuro.

### **Gestión del Tiempo en Actividades Remotas**

La “gestión del tiempo”, o en su defecto el “manejo del tiempo”, podría calificar como predictor del rendimiento académico, en tanto es una estrategia que posibilita que el estudiante se autorregule. No obstante, a partir del contexto de “Pandemia” (2020) dada la situación epidemiológica ante el virus “COVID-19” que impactó a nivel mundial, se incorporó con más fuerza el uso de las “nuevas tecnologías de información y comunicación” (NTIC) en el campo de la educación. Por ello, en palabras de Sánchez-Calvo, L; Alvarenga-Venutolo, S (2015), *“los modelos pedagógicos están en crisis. De esquemas lineales, autoritarios, analógicos, se está pasando a modelos en red, participativos y digitales. La unidisciplinariedad es sustituida por la convergencia e interacción entre múltiples disciplinas. La enseñanza y el aprendizaje “in situ” quedan atrás frente a las alternativas de trabajo virtual y compartido” (La virtualidad en los procesos educativos. Tecnología en marcha; p,123. Vol,28 N°1)*

A modo de ejemplificar lo dicho, se propone considerar el desafío titulado *“Estrategias de evaluación formativa en la enseñanza de programación en modalidad remota de emergencia” (2020)*. Desarrollado por el equipo docente de la asignatura “Algoritmos y Estructuras de Datos I”, de primer año de la carrera “Licenciatura en Sistemas de Información” de la Universidad Nacional del Nordeste. Quienes, en un escenario de distanciamiento físico del estudiante, decidieron adecuar su estructura de enseñanza presencial a una modalidad de “enseñanza remota de emergencia”. Es decir, la situación implicó tomar decisiones pedagógicas sobre cómo abordar el proceso de enseñanza-aprendizaje. Para ello, han tenido que elaborar nuevos materiales y nuevos métodos de evaluación. En efecto, en este escenario de virtualidad, se promovió aún más el aprendizaje activo y la motivación. Aproximadamente un 41% de los estudiantes que

iniciaron el dictado remoto (2020), lograron cumplir las condiciones para aprobar la materia.

Sin dudas, los espacios virtuales de aprendizaje favorecen aspectos diferentes a los espacios presenciales. Uno de ellos, tiene que ver con superar la rigidez de los horarios y moverse con flexibilidad en la distribución y organización del tiempo de estudio, al combinarlo con otras ocupaciones.

Saber manejar y planificar el tiempo son entonces, comportamientos trascendentales de impacto positivo en el proceso de aprendizaje. Es fundamental poner atención en la capacidad de *autorregulación* entendiéndose como “*acciones, sentimientos y pensamientos autogenerados que son planeados y cíclicamente adaptados para alcanzar metas de aprendizaje*” (Zimmerman, 2000; Zimmerman et al., 2005). Es decir, tanto en ámbitos presenciales como virtuales, es deseable que el estudiante se posicione desde un rol activo en la construcción del conocimiento. La diferencia reside en que, en escenarios híbridos, el tiempo podría ser vivenciado de modo laxo, flexible, debido a la falta de estructura de la virtualidad. En consecuencia, el estudiante debe adoptar con más énfasis comportamientos tales como la planificación de las actividades, la regulación del esfuerzo, control del desempeño en las actividades, el manejo del ambiente de estudio, entre otros.

Por lo anterior, cabe mencionar que estudiar requiere no sólo tiempo disponible para hacerlo, sino manejar ciertas habilidades y predisposición para sostener el proceso de enseñanza-aprendizaje. Pues, aprender es una tarea que requiere esfuerzo, tanto en la cantidad de tiempo que se le dedica, sino también en la calidad del tiempo que conlleva dicho esfuerzo.

## **2. Objetivo**

Partiendo del planteo precedente, el presente trabajo se propone conocer la gestión del tiempo en actividades remotas, complementarias a la modalidad de aprendizaje presencial, de estudiantes ingresantes a la carrera de Licenciatura en Sistemas de la FaCENA – UNNE. Ello permitirá la descripción e interpretación de la percepción de los estudiantes respecto a cómo transitan el aprendizaje; teniendo en cuenta que actualmente es posible complementar los escenarios de aprendizaje presencial con actividades en modalidad remota.

### **2.1. Instrumento**

La información fue recolectada mediante un cuestionario de 20 ítems que varían entre preguntas abiertas y cerradas. Éste fue elaborado mediante la herramienta *Google Forms*, elegida por su facilidad para elaborar, distribuir y aplicar. El mecanismo para esto último consistió en enviar el formulario vía correo electrónico a todos los estudiantes de primer año de la Licenciatura en Sistemas de la información de FaCENA – UNNE que han participado del curso de ingreso denominado “Ambientación a la Universidad”, realizado por el “Gabinete Psicopedagógico” de la FaCENA.

### **2.2. Sujetos**

El grupo que participó, de manera voluntaria, completando el formulario, está constituido por 44 estudiantes. Éstos cursan el primer año de la carrera mencionada con anterioridad. A continuación, una breve descripción que caracteriza los participantes, obtenida del instrumento aplicado:

La edad, si bien es variada, es posible observar que la mayoría de los participantes dicen tener entre 18 y 19 años (Fig. 1).

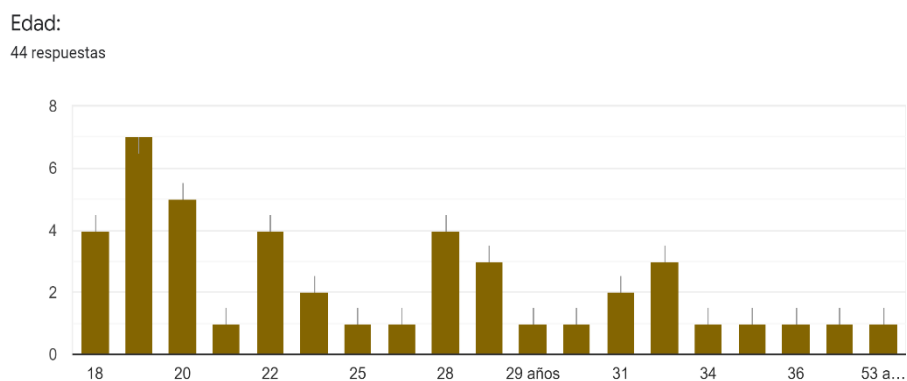


Fig 1. Grupo etario de participantes encuestados.

En cuanto al número de las materias que actualmente cursan, el gráfico muestra que existe la misma cantidad de estudiantes cursando entre 1 materia y 2 materias del primer año de la carrera. (Fig. 2). Cabe mencionar que en cuanto a la estructura curricular el plan de estudio de esta carrera posee 5 asignaturas en su primer año las cuales se dividen en dos en el primer cuatrimestre ( Algoritmos y Estructuras de Datos I y Algebra) y 3 en el segundo cuatrimestre (Algoritmos y Estructuras de Datos II y Logica y matematica computacional y Sistemas y Organizaciones).

Cabe destacar que, con el propósito de favorecer la inserción del alumno en la vida universitaria, la FaCENA implementó estrategias superadoras a nivel de la carrera. Por ello, en el año 2009, se modificó el plan de estudios de la LSI con efecto desde el primer año en el 2010. Éste establece sólo dos asignaturas en el primer cuatrimestre; esta distribución, se orientó a facilitar la inserción y adaptación del estudiante en la vida universitaria al permitir una mayor concentración en esas asignaturas, en la búsqueda de un mejor desempeño y construcción de hábitos de estudio.

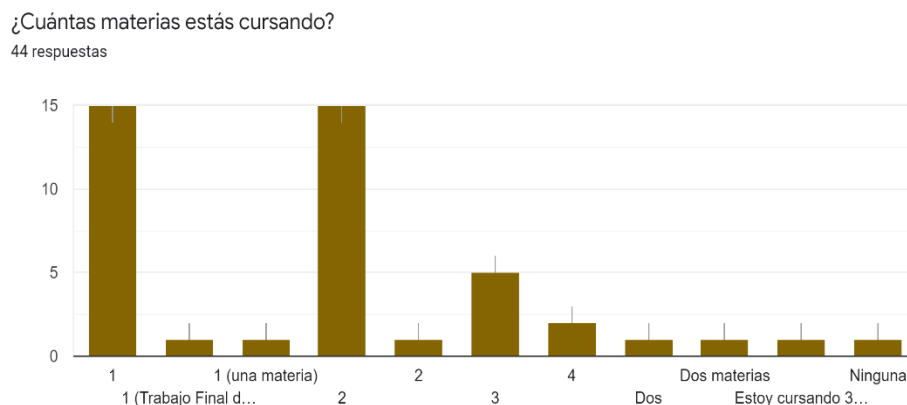


Fig. 2 Cantidad de materias que los estudiantes cursan actualmente

No obstante, un dato que resulta significativo es el obtenido ante la identificación de género. El 75% de los participantes, de este instrumento, se perciben mujer. Por lo que, ¿habría modificación, en la gestión del tiempo, según posibles actividades diferenciadas entre hombre y mujer? (Fig. 3).

Al respecto, dicho porcentaje, alienta a estudiar la cuestión de género tanto en la LSI como en carreras CTIM (Ciencia, Tecnología, Ingeniería, Matemáticas) en general. Dado que, ante el cambio tecnológico y su influencia en los sistemas productivos, hombres y mujeres deben contar con las capacidades y habilidades necesarias para hacer frente a la revolución tecnológica. De este modo, el sistema educativo contribuiría a eliminar la brecha de género.

Te autopercibis:  
44 respuestas

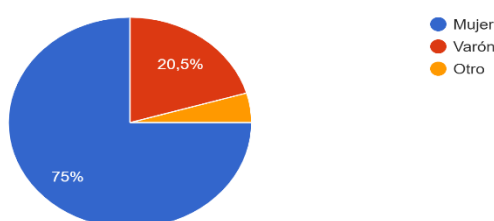


Fig.3 Porcentajes de autopercepción de los estudiantes.

Asimismo, el 75% de los estudiantes respondieron que viven con sus familias. Más allá de desconocer cómo se conforman dichas familias, la respuesta se torna interesante para pensar ¿será posible que la convivencia familiar influya en el manejo del tiempo? O, tal vez, dicho porcentaje indica que hay que tener en cuenta que ¿no sólo el “alumno estudia”, sino que su tiempo debe dedicar a tareas del hogar? (Fig. 4).

Vivís con:  
44 respuestas

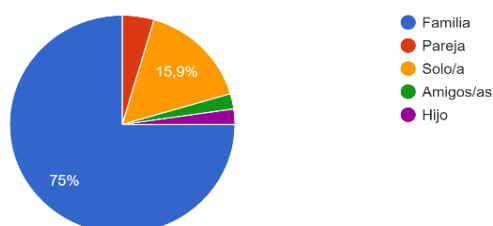


Fig.4 Porcentajes de respuestas que indican con quienes viven los estudiantes

### 3. Resultados

Tras indagar sobre el “estudio en modalidad remota”, el 56,8% de estudiantes han respondido que sí lo hacen; en contraposición al 43,2% que no. No obstante, vale aclarar

que dicha modalidad es optativa, ya que su uso complementa la modalidad de aprendizaje presencial que actualmente se sostiene en la FaCENA. (Fig.5)

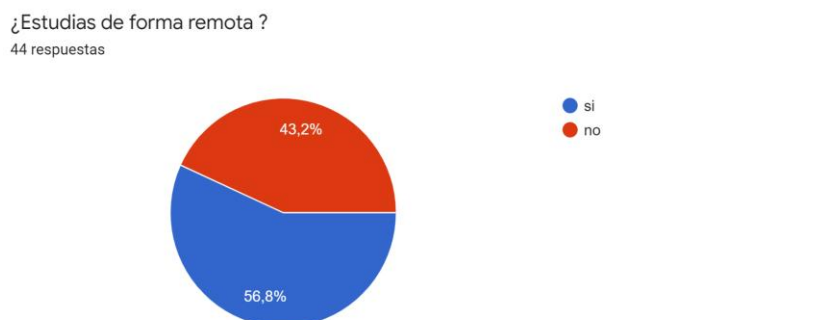


Fig.5 Porcentaje de alumnos que estudian en forma remota

Por otro lado, el tiempo que los estudiantes requieren para sus actividades en modalidad remota dice ser entre 2 a 4 horas diarias. (Fig. 6).

Al respecto, se presupone que dichas horas, serían utilizadas en la implementación de estrategias de estudio actuales. Como ser, el encuentro en línea con pares para lecturas obligatorias; utilización del canal de YouTube para la búsqueda de ejemplos prácticos; realización de autoevaluación en el aula de la plataforma “Unne Virtual”; entre otros. Pero, conviene subrayar que, entre 2 a 4 horas diarias, sería insuficiente para cumplir con las exigencias de una carrera de grado. Sin embargo, la mayoría de los ingresantes provienen de la educación media, por lo que habría que interrogar si es a consecuencia de la flexibilización en las exigencias durante la Pandemia 2020.

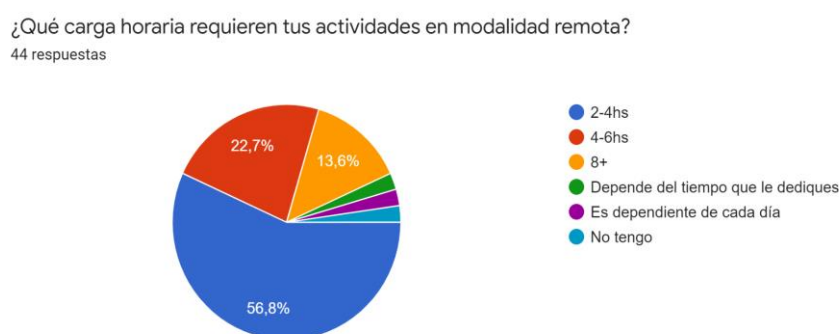


Fig.6 Carga horaria diaria requerida en actividades de modalidad remota

Además, tomando como antecedente la “enseñanza remota de emergencia” implementada en Pandemia (2020), se preguntó acerca de los desafíos de asistir a clases en modalidad virtual. Obteniendo como resultado que el 31,8% tiene dificultades a la hora de concentrarse; el 22,7% para gestionar y organizar su tiempo y el 20,5% respondió que su desafío es descartar la opción de “mirar las clases grabadas”. (Fig. 7).

A la hora de asistir a clases en modalidad virtual ¿Cuál es tu mayor desafío?  
44 respuestas

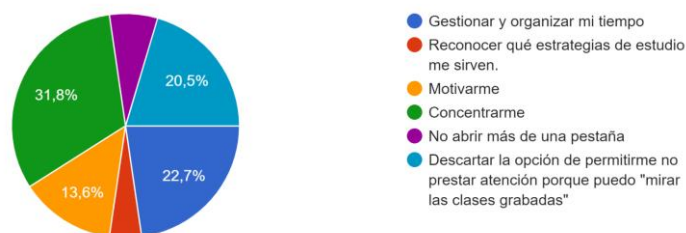


Fig.7 Desafíos ante la asistencia a clases en modalidad virtual

Por otro lado, frente a las actividades remotas, en este caso “estudiar en modalidad virtual”, el 43,2% logra hacerlo hasta que se distrae, asintiendo que luego le cuesta retomar; mientras que el 36,4% lo realiza hasta terminar la actividad y sólo el 20,5% destina un tiempo, para la realización de la actividad, determinado con anticipación. (Fig. 8).

Cuando te propones trabajar o estudiar de manera remota lo haces:  
44 respuestas

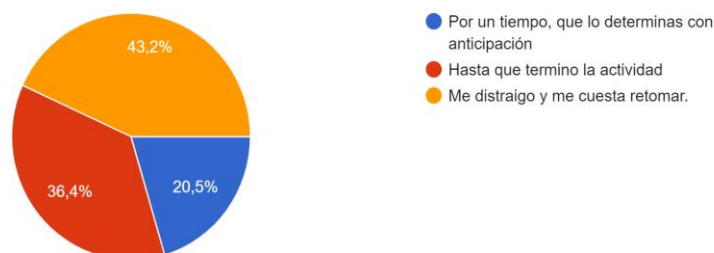


Fig.8 Porcentajes que indican cómo se proponen estudiar desde lo remoto

Partiendo de la respuesta anterior, frente al porcentaje elevado de estudiantes que respondieron que se distraen, se evaluó las razones que podrían tenerse en cuenta sobre ello. Como resultado, al preguntar qué actividad interrumpe la gestión de su tiempo, el 47,7% de los estudiantes hace alusión a distractores tales como redes sociales, celular, juegos, entre otros. Mientras que el 38,6% refirió a las tareas hogareñas/ personas a cargo/ materno/ paterno. Es importante resaltar que de este 38,6%, el 99% son mujeres. (Fig. 9).

Selecciona qué opción consideras que interrumpen la gestión del tiempo:  
44 respuestas

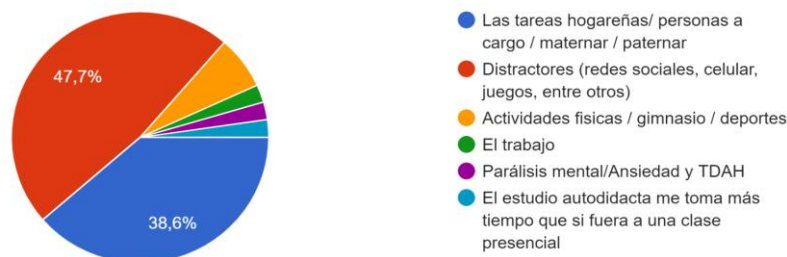


Fig. 9 Distractores que interrumpen la gestión del tiempo

En consecuencia, el 36,4% de los participantes opinó que una de las habilidades blandas que requieren las actividades en modalidad remota, es la adaptabilidad; y el 27,3% la comunicación. (Fig.10)

¿Qué habilidad blanda consideras que requieren las actividades en modalidad remota?  
44 respuestas

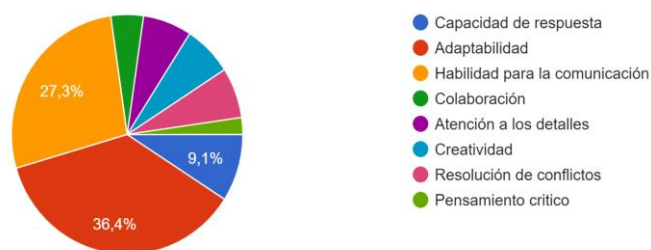


Fig.10 Habilidades requeridas para la modalidad remota

Por último, el 95,5% de los estudiantes consideran que la Universidad debe enseñar a gestionar el tiempo. (Fig.11)

¿Consideras que la universidad debería proporcionar herramientas que enseñen a gestionar el tiempo?  
44 respuestas

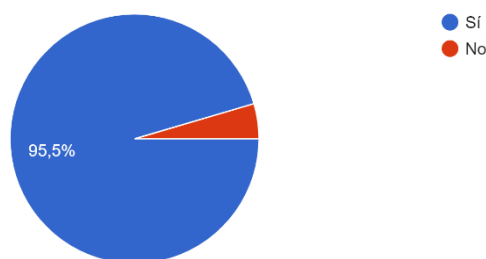


Fig.11 Enseñanza de gestión del tiempo en la Universidad

### 3.1. Algunas acciones en torno a la gestión del tiempo

A partir de las demandas recibidas de los estudiantes desde el año 2018 hasta la actualidad, desde el espacio del gabinete psicopedagógico se han elaborado diferentes talleres en relación a la “gestión y organización del tiempo”, como ser: “Herramientas para la organización y gestión del tiempo”; “El estrés académico, estrategias para su manejo”; “Caja de herramientas emocionales” y “Preparar un examen”. El más actual fue “Taller de gestión del tiempo” llevado a cabo el 7 de mayo del año 2022. En el mismo, se brindó herramientas y estrategias para mejorar y potenciar la buena administración, gestión y organización del tiempo académico. Con un total de aproximadamente 60 participantes; la recepción al taller fue sumamente favorable y en el mismo se evidenció la demanda y los desafíos que enfrentan los estudiantes para lograr distribuir sus tiempos y sus actividades.

Antes de la realización del taller de Gestión del tiempo, se recopilaron las principales inquietudes de los estudiantes:

Nos mencionas brevemente ¿Cómo te sentís siendo estudiante universitario? ¿Qué es lo que más te cuesta en tu organización personal? ¿Cómo te gustaría que el gabinete te acompañe?, ante esta pregunta algunas de las respuestas de los estudiantes fueron las siguientes:

*-Me siento con ganas de aprender ya que es algo que me gusta, pero a pesar de ello no administro eficientemente mi tiempo como para darle la prioridad que quisiera. Me gustaría conocer alguna forma que ayude a gestionar mejor mis actividades.*

*-Me cuesta mucho cumplir las tareas que me asigno y me distraigo mucho.*

*-No sé cuántas horas dedicar a cada materia y me cuesta organizarme para llegar a estudiar todos los temas para los parciales, casi nunca llego.*

*-Bastante mal, tengo que aprender a organizarme y poner toda mi atención en el estudio, cosa que me cuesta ya que venía de una secundaria donde estudiaba unao 2 horas y ya aprobaba el examen y más de la cuarentena, donde todo era súper sencillo y aprobabas con menos del mínimo esfuerzo, cosa que parece no ser así acá porque no aprobé ninguno de los parciales, algunos por mas otros por menos diferencia pero no aprobé.*

*-Necesito ayuda para administrar mi tiempo para poder estar más al día y sentirme más a gusto con la carrera*

*-Me cuesta la organización de tiempo y motivación al estudiar*

*-Me siento mal, no sé cómo organizarme para estudiar*

*-Siento que no sé por dónde empezar y tengo tantas cosas que hacer, respecto a la universidad como también a lo personal, por eso mismo me interesa este taller y espero que me pueda ayudar.*

*-Me está costando mucho adaptarme y sobre todo organizar mis tiempos, me concentro demasiado en una sola materia y pierdo el hilo en las otras*



*-Como no estoy acostumbrada al estudio diariamente, duermo mal, no cumplo con las horas de sueño, hay veces que no duermo en toda la noche por estudiar y al otro día no rindo bien físicamente, no sé por dónde ni por qué materia empezar a estudiar, tampoco puedo terminar de organizarme bien con las ellas.*

*-Es la mejor etapa que estoy viviendo, pero aun así me faltan estrategias para organizar productivamente mi tiempo de estudio y en eso estoy interesado.*

*-Como estudiante universitario me siento todo el tiempo sin tiempo, valga la redundancia. Me cuesta mucho organizarme con eso.*

Por consiguiente, a través de las respuestas que brindan, se evidencia que no saber planificar las actividades y el manejo del tiempo que éstas conlleva, produce impacto en el rendimiento académico y en la “representación del ser estudiante” que cada sujeto podría internalizar de sí mismo. Pues, en palabras de estudiantes, “Necesito ayuda para administrar mi tiempo para poder estar más al día y sentirme más a gusto con la carrera”; “Me está costando mucho adaptarme y sobre todo organizar mis tiempos, me concentro demasiado en una sola materia y pierdo el hilo en las otras”; “Me siento mal, no sé cómo organizarme”;

Analizando estas respuestas, se evidencia que, la falta de organización del tiempo produce malestar e insatisfacción en cuanto al rendimiento académico. Sumado a ello, la incertidumbre del paso a un ciclo de formación profesional como lo son los estudios universitarios, Por tanto, cabe destacar la relevancia que implica el trabajo coordinado entre las diferentes áreas de la FaCENA, dar a conocer los resultados y acciones, poner en práctica estrategias que coadyuven a los estudiantes en estos aspectos, dejar plasmados lineamientos de trabajo, y finalmente continuar reforzando otras habilidades blandas a efectos de mejorar el rendimiento académico de los estudiantes. En efecto, para evitar la lectura de estos factores de manera aislada, sugerimos que el eje sea la responsabilidad institucional compartida, en tanto se reconoce que los factores que inciden en el rendimiento académico tienden a ser tanto endógenos (ubicado en el estudiante, como ser, desconocer las formas de gestionar el tiempo) como exógenos (ubicado en el mundo circundante).

### **3.2 Algunas habilidades blandas a desarrollar:**

En la actualidad, se evidencian cambios tecnológicos a velocidad exponencial. Reflejando un rol central en la reconfiguración de las formas de producción y consumo. De modo que, la revolución digital, generó nuevos modelos de negocios, transformó las comunicaciones y alteró las formas de empleo. Frente a lo cual, la Universidad, debe favorecer a los estudiantes la adquisición de aptitudes, ya que el sistema educativo es responsable de la formación de profesionales competitivos.

En este sentido, en mira al futuro laboral de estudiantes de “Licenciatura en Sistemas”, aprender a optimizar la gestión del tiempo y a desarrollar habilidades blandas, favorece el transitar académico y la futura adaptación a la inserción laboral. Puesto que, en la actualidad, no se valoran solamente conocimientos técnicos, sino que se privilegian las habilidades blandas. Por este motivo, el Gabinete Psicopedagógico, concretó encuentros con estudiantes bajo la modalidad de taller: “Caja de herramientas

emociones”; “Cómo prepararse para un examen” (julio 2022). A continuación, algunas habilidades blandas trabajadas:

**A- Adaptabilidad:** capacidad de responder adecuadamente a las exigencias del entorno. Permitiendo que se desarrolle en simultáneo la habilidad de autogestión, ya que, para emprender, no se necesita de la solicitud de un agente externo. Por el contrario, se realizan tareas porque se sabe o se entiende que se necesitan.

**B- Comunicación:** Florencia & Fontán en “*Habilidades blandas en ingeniería de software*” (2015) define como la capacidad de expresar su punto de vista, de contribuir objetivamente a discusiones y de dar y recibir retroalimentación. A la vez, para recibir con la misma importancia, tanto las opiniones ajenas como las propias. Por último, favorece la tolerancia.

**C- Capacidad de respuesta:** ésta se articula con la capacidad de aprender a tomar decisiones. Ello se trata de elegir de manera eficaz y constructiva las acciones a realizar ante distintas situaciones y contextos de la vida cotidiana. Se espera que se genere un proceso a partir del cual las decisiones sean a partir de la razón y no por impulso.

#### 4. Conclusiones

Frente a los resultados hallados, se concluye la significatividad del contexto educativo en modalidad virtual como una experiencia productiva, si se realiza de manera adecuada. Para ello, es fundamental la gestión del tiempo, repercutiendo en el proceso de aprendizaje al asumir una posición proactiva. . Pues, se considera que, por las características que imprimen los espacios virtuales, pudiendo ser atemporal, flexible, autónomo, resultan aspectos complejos para que el estudiante alcance cierta planificación y capacidad de autorregulación. No obstante, dicha complejidad aspira a que sea significada como transitoria, ya que indica falta de hábito o desconocimiento. Asimismo, hay que destacar que, dicha complejidad, también podría verse acentuada al ser el primer año de trayectoria universitaria. Es decir, los estudiantes frente a ello, se topan con cambios no sólo en la responsabilidad que implica sobrellevar la exigencia de una carrera de grado sino, también, con las interacciones asincrónicas entre profesor-estudiante; la implementación de nuevas estrategias cognitivas; la motivación académica; el manejo del ambiente de estudio; entre otros.

En este orden de ideas, como gabinete psicopedagógico de la FaCENA-UNNE consideramos pertinente descartar la unidisciplinariedad para dar lugar al trabajo articulado con docentes. El cual, una posible línea de trabajo compartido, constituye el desarrollo de “habilidades blandas” en conjunto con las “habilidades duras” propias del saber académico, conceptual, que se espera obtener de la carrera en cuestión. Recordemos que para la RAE (Real Academia Española), “habilidad” significa “capacidad para hacer”, por lo que se trata de enseñar a tener la capacidad para gestionar el tiempo, el comportamiento, la comunicación, entre otros. De modo que, los estudiantes de la FaCENA, no solamente se beneficien mediante la adquisición de saberes de índole científico y disciplinar; por el contrario, los resultados obtenidos visualizan la necesidad de instalar una formación educativa integral que acompañe la realización plena en la inserción a la vida cotidiana, académica y laboral, tanto para hombres como para mujeres.

En consecuencia, dada las respuestas obtenidas por los estudiantes, quienes opinan que el mayor desafío sobre la virtualidad, como espacio de aprendizaje, es la concentración; la adaptabilidad como habilidad necesaria para dicha actividad y la preferencia por obtener de la Universidad enseñanzas respecto al manejo del tiempo, sugerimos las siguientes líneas de trabajo articulado con docentes y/o departamentos académicos:

- Reforzar las competencias digitales para el tránsito en la Universidad. Ello permitirá que los estudiantes conozcan la capacidad, el potencial y la limitación del entorno virtual que utilizan. Además, dichas competencias, repercutirá en la organización, el tiempo y la confianza de cada estudiante sobre el uso de dicho espacio.
- Promover el sentido de pertenencia hacia la Universidad. Se espera, por medio de ello, que los estudiantes aniden sentimientos, se identifiquen, a fin de encontrar en los espacios físicos, virtuales y en convivencia con otros, la motivación académica para permanecer como estudiantes. Y, la confianza para aprender a pedir colaboración tanto a pares como a docentes.
- Instalar talleres cuyos espacios de participación faciliten el complemento de las “habilidades duras” a partir del desarrollo de las “habilidades blandas”. Como ser, la habilidad para comunicar, resolver conflictos, ejercer liderazgos, trabajar en equipo. Estas habilidades, de carácter “blandas”, deben ser incorporadas de modo que beneficien la inserción a la práctica laboral. Pues, como casa de altos estudios, habrá que encauzar los esfuerzos para la formación en aspectos cognitivos sin dejar de lado los aspectos sociales.
- Insistir en nuevas líneas de investigación, a partir de lo aquí descrito. Se podrá tomar en consideración el porcentaje de mujeres que estudian la carrera de Lic. en Sistemas, para indagar cómo distribuyen el tiempo entre lo académico y la cotidianidad en general. De este modo, se propone indagar para ampliar el campo de conocimientos, desde una perspectiva de género, para retroalimentar los datos y estar a la altura de la subjetividad de la época. Por otro lado, la relevancia de considerar los datos arrojados por intermedio del instrumento para, a partir de la lectura de ello, ahondar el trabajo con la finalidad de repensar el impacto en el rendimiento académico.

## 5. Referencias Bibliográfica:

- Comisión Económica para América Latina y el Caribe (CEPAL). “La autonomía de las mujeres en escenarios económicos cambiantes” (LC/CRM.14/3), Santiago, 2019.
- Chiecher, Analía; Donolo, Danilo; Rinaudo, María Cristina. (2009) “Regulación y planificación del estudio. Una perspectiva comparativa en ambientes presenciales y virtuales”. En revista “Electronic Journal of Research in Educational Psychology”, vol. 7, núm. 1, abril, 2009, pp.209-224 Universidad de Almería. Almería, España.
- Chiecher, Analía, Donolo, Danilo y Rinaudo, María Cristina (2009).Gestión del tiempo y el ambiente de estudio en cursos online. Estudio comparativo de tres

grupos de estudiantes de diferente perfil. Congreso Internacional de Investigación y Práctica Profesional en Psicología XVI Jornadas de Investigación Quinto Encuentro de Investigadores en Psicología del MERCOSUR. Facultad de Psicología. Universidad de Buenos Aires, Buenos Aires.

- Dapozo, G; Greiner, C; Godoy Guglielmone, María V. “Retención de alumnos y culminación de estudios. Análisis y propuestas para la Licenciatura en Sistemas de Información de la UNNE”.
- Dapozo, G; Greiner, G; Petris, R y otros. (2020). “Estrategias de evaluación formativa en la enseñanza de programación en modalidad remota de emergencia”. XXVI Congreso Argentino De Ciencias De La Computación (CACIC 2020).
- De la Barrera, María Laura; Donolo, Danilo Silvio; Rinaudo, María Cristina. (2008). “Ritmo de estudio y trayectoria universitaria”. En Revista “Anales de Psicología”, vol. 24, núm. 1, junio, 2008, pp. 9-15. Universidad de Murcia. Murcia, España.
- Fernandés, M; y otros. (2019). “Diseño Instruccional para fortalecer las competencias digitales en
- ingresantes: el caso de la FaCENA-UNNE”. XXV Congreso Argentino de Ciencias de la Computación. Facultad de Ciencias Exactas y Naturales y Agrimensura. Universidad Nacional del Nordeste. Corrientes– Argentina
- Lester V. Jerez. (2020). “Habilidades blandas deseadas en profesionales de TI por empleadores de la empresa privada y pública costarricense”. En Revista Investiga TEC, septiembre 2020, pp.17-20. Costa Rica.
- Sánchez-Calvo, L; Alvarenga-Venutolo, S. “La virtualidad en los procesos educativos: reflexiones teóricas sobre su implementación”. *Tecnología en Marcha*. Vol. 28, N° 1, Enero-Marzo. Pág 121-129. Costa Rica.
- Porcel, E., Dapozo, G. y López, M. (2010). Predicción del rendimiento académico de alumnos de primer año de la FACENA (UNNE) en función de su caracterización socioeducativa. *Revista Electrónica de Investigación Educativa*, 12(2).

## **Evaluación de habilidades de pensamiento computacional al inicio de una asignatura de programación en una carrera de Informática**

Gladys Dapozo, Cristina Greiner, Raquel Petris, Emanuel Irrazabal, Ana María Company, María Cecilia Espíndola, Yanina Medina

Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura  
Universidad Nacional del Nordeste (UNNE), Av. Libertad 5450, 3400, Corrientes, Corrientes,  
Argentina

{gndapozo, cgreiner, rpetris, eirrazabal, acompany, mcespindola, yanina}@[exa.unne.edu.ar](mailto:exa.unne.edu.ar)

### **Resumen**

En este trabajo se presentan los resultados de la evaluación de habilidades de pensamiento computacional, tomadas de un test diagnóstico realizado al inicio de una asignatura de introducción a la programación. Los estudiantes que participaron de esta experiencia son ingresantes de una carrera de Informática. El dictado de esta asignatura ha regresado a la presencialidad plena y al inicio del curso se les tomó un test diagnóstico con actividades de resolución de problemas que los alumnos tenían que realizar en forma intuitiva, sin ninguna indicación adicional, más que el planteo de la actividad. Adicionalmente se recabó información acerca de la experiencia previa que tenían en programación. Los resultados obtenidos permitirán consolidar estrategias de enseñanza que se enfoquen en la adquisición de habilidades de pensamiento computacional y enfatizar la idea de que la programación no es una actividad meramente técnica, sino que requiere un conjunto de habilidades combinadas de resolución de problemas.

**Palabras clave:** Habilidades del Pensamiento Computacional. Enseñanza de la programación. Carreras de Informática

### **Introducción**

#### **Enseñanza de la programación**

Diversos estudios [1] [2] [3] señalan que es evidente la producción limitada de graduados en ciencias, tecnología, ingeniería y matemática respecto de las necesidades del aparato productivo y, más significativamente, la falta de masa crítica en capacidades necesarias para

trabajar en la innovación digital, entre las que destaca la carencia estructural de capital humano de TIC (Tecnologías de la Información y la Comunicación).

Las carreras de informática contribuyen con la formación de estos recursos, sin embargo, enfrentan diversas dificultades que conducen a una insuficiente producción de profesionales. Entre ellas, se encuentra el desgranamiento temprano, particularmente en los primeros años de la universidad. En este periodo se inicia la enseñanza de la programación, siendo esta un área de estudio con la que la mayoría de los estudiantes no ha tenido un contacto previo, y, por tanto, el aprendizaje conlleva un mayor grado de dificultad [4]. A la hora de planificar el dictado de la programación inicial es común el planteo de diversos interrogantes sobre ¿cómo se enseña?, ¿por dónde se empieza?, ¿qué paradigma se debe utilizar?, ¿qué lenguaje de programación se debe emplear?, ¿cómo se deben orientar los ejercicios?, entre otras cuestiones [5].

En [6] se señala que la investigación existente ha llevado a muchas discusiones e ideas sobre la mejor manera de enseñar la programación introductoria, y de las dificultades que enfrentan los estudiantes en este tipo de cursos. Entre estas, el hecho de que los estudiantes consideran a la programación como una actividad puramente técnica en lugar de un conjunto de habilidades combinadas de resolución de problemas. Por lo tanto, la mayoría de los estudiantes que se inician en la programación tiende a desarrollar un conocimiento superficial y no crean estrategias de resolución de

problemas mediante el uso de construcciones de programación.

En [7] se propone una estrategia de enseñanza de programación en estudiantes universitarios que consiste en el desarrollo de un conjunto de actividades orientadas a estimular el pensamiento computacional mediante herramientas lúdicas con el objetivo de motivar a los alumnos y facilitar el aprendizaje de la programación. Los resultados indican que estas actividades contribuyeron positivamente a incrementar la motivación de los alumnos e incorporar una metodología de resolución de problemas que facilita la comprensión de los conceptos básicos de programación que forman parte de los contenidos de la asignatura.

### **Pensamiento computacional**

Jeannette M. Wing, una de las primeras especialistas en utilizar el término de pensamiento computacional, lo definió de la siguiente manera: “el pensamiento computacional es el proceso de pensamiento involucrado en la formulación de problemas y sus soluciones para que las soluciones estén representadas en una forma que pueda ser efectivamente llevada a cabo por un agente de procesamiento de información” [8].

En otras palabras, se trata del proceso mental a través del cual una persona, planteado un problema, para su posible solución utiliza una secuencia de instrucciones ejecutadas por una computadora, un humano o ambos.

En [9] los autores señalan que el pensamiento computacional es un proceso cognitivo que implica el razonamiento lógico por el cual se resuelven los problemas, y los procedimientos y sistemas se entienden mejor. Abarca:

1. La capacidad de pensar de forma algorítmica: El pensamiento algorítmico es la capacidad de pensar en términos de secuencias y reglas como una forma de resolver problemas. Es una forma de llegar a una solución a través de una definición clara de los pasos.
2. La capacidad de pensar en términos de descomposición: La descomposición es una manera de pensar un problema en términos de sus partes y componentes. Cada parte debe entenderse, solucionarse, desarrollarse y

evaluarse por separado. Esto hace más fácil de resolver problemas complejos, y que los grandes sistemas sean más fáciles de diseñar. Luego las partes se integran para obtener la solución completa del problema.

3. La capacidad de pensar en generalizaciones, identificando y haciendo uso de patrones: La generalización se asocia con la identificación de patrones, similitudes y conexiones, y la explotación de las características similares. Es una forma de resolver rápidamente los nuevos problemas sobre la base de las soluciones de los problemas anteriores, y la construcción en experiencias previas.
4. La capacidad de pensar en términos abstractos, la elección de buenas representaciones: La abstracción es el proceso de hacer un problema más comprensible a través de la reducción de los detalles innecesarios. Una vez determinadas las características relevantes, se crea un “modelo” o representación del problema, que brinda una idea general del problema que se intenta resolver. Una parte fundamental de la abstracción es la elección de una buena representación de un sistema.
5. La capacidad de pensar en términos de evaluación: habilidad para determinar la eficacia y la eficiencia en el uso de recursos.

Lo mismos principios propone Selby [10] como habilidades de Pensamiento Computacional, estableciendo también criterios para desarrollarlos

- Generalización: la habilidad para expresar la solución de un problema en términos genéricos, la cual pueda ser aplicada a diferentes problemas que comparten características como el problema original.
- Descomposición: fraccionar a piezas más pequeñas, fáciles de resolver, partes de un problema.
- Abstracción: habilidad para decidir qué detalles de un problema son importantes y qué detalles se pueden omitir.
- Diseño Algorítmico: habilidad para crear un conjunto de instrucciones que indiquen paso a paso la solución de un problema.
- Evaluación: habilidad para reconocer y determinar los alcances de realizar procesos, en términos de eficiencia y uso de recursos.

El pensamiento computacional está relacionado con otros tipos de pensamiento: el matemático, el lógico y el crítico, entre otros, con los cuales comparte habilidades cognitivas comunes, como ser, reconocimiento de patrones, abstracción, modelado, repetición, entre otras. El pensamiento computacional favorece, a partir del reconocimiento de los aspectos que nos rodean, de problemas reales de las actividades diarias o de las ciencias, la propuesta de soluciones aplicando herramientas informáticas [11].

En [12] se ha realizado una evaluación de habilidades específicas del pensamiento computacional en estudiantes de nuevo ingreso de una carrera de tecnologías de la información con el objetivo de favorecer el desempeño académico de los estudiantes. Se seleccionaron cinco reactivos en correspondencia con las habilidades del pensamiento computacional (abstracción, generalización, descomposición, diseño algorítmico y evaluación) y establecieron una relación con los contenidos temáticos del curso, en base a lo cual determinaron fortalezas y debilidades de los estudiantes respecto de las habilidades del pensamiento computacional.

Por otra parte, la ciencia se desarrolla, crece y avanza por el creciente influjo de la Informática, que contribuye al manejo de cantidades masivas de datos permitiendo extraer conclusiones relevantes en cada área. Este fenómeno se observa aún en ciencias que hasta hace pocos años se habían mantenido al margen de estos desarrollos tecnológicos, como ser las ciencias sociales. El autor propone que, por su aporte a las herramientas mentales disponibles, es necesario incluir los fundamentos computacionales de la Informática entre las ciencias básicas en todos los niveles educativos. El pensamiento computacional representa así un componente irremplazable de las habilidades necesarias para comprender y desempeñarse en el mundo [13].

Todo esto evidencia la necesidad de alcanzar en los estudiantes la comprensión del uso de la tecnología informática, no como un simple requisito para aprender un lenguaje de programación mediante la elaboración de algoritmos y la generación de código, sino que

se busca la apropiación de un proceso metodológico adecuado para resolver problemas y programar.

Basado en los conceptos expuestos, en la asignatura Algoritmos y Estructuras de Datos I de la carrera Licenciatura en Sistemas de Información de la Facultad de Ciencias Exactas y Naturales y Agrimensura de la Universidad Nacional del Nordeste (UNNE), se propuso a los estudiantes la resolución de un conjunto de actividades vinculadas con las habilidades del pensamiento computacional propuestas por Selby [10], con el objetivo de detectar tempranamente los déficits y proponer actividades que refuercen esas habilidades. También se consideraron otras variables como por ejemplo la experiencia previa en programación y el género, a fin de relacionar el perfil del estudiante con sus habilidades computacionales, como una información más a tener en cuenta.

## Metodología

Al inicio del dictado de la asignatura Algoritmos y Estructuras de Datos I (marzo 2022), se propuso a los estudiantes que concurrían a la primera clase de contenidos teóricos que respondieran un cuestionario impreso con preguntas acerca de su experiencia previa en programación y la resolución de 5 actividades problemáticas, que tenían que resolver auxiliados de papel y lápiz, y el resultado volcarlo al cuestionario. Se les pidió que resolvieran las actividades intuitivamente, sin darles ninguna indicación adicional.

A continuación, el diseño del cuestionario:

### 1. **Experiencia previa en programación** (Marcar el ítem (uno solo) que considere más representativo)

- 1.1. ¿Tuviste alguna experiencia de programación antes de ingresar a la Facultad?
  - a. Si, en la escuela secundaria o primaria
  - b. Si, estudiando por mi cuenta
  - c. Si, en otra carrera (universitaria o terciaria)
  - d. No, ninguna experiencia previa

- 1.2. Si la respuesta anterior fue 1.1.a, ¿qué tipo de actividades realizaste?
- Programación con PilasBloques
  - Programación con Scratch
  - Otras herramientas de programación de tipo lúdica
  - Programación con lenguajes de programación (C, Java, Python, otros)

**2. Test de habilidades de pensamiento computacional**

Se consideraron 5 actividades que evalúan las habilidades de pensamiento computacional propuestas por Selby [10]:

- 1-Abstracción: Canguro
- 2-Diseño Algorítmico: Castores en movimiento
- 3-Generalización o detección de patrones: Espías
- 4-Evaluación: Salto de charcos
- 5- Descomposición: Móviles

2.1. **Canguro.** Hay 10 platos en una fila. Hay una manzana en cada plato. Al canguro Tomás le encanta saltar. Primero, el salta desde el plato más a la izquierda con la letra A. En cada salto después de la inicial, salta dos platos hacia adelante, o tres platos hacia atrás (ver ejemplo de la Fig. 1).

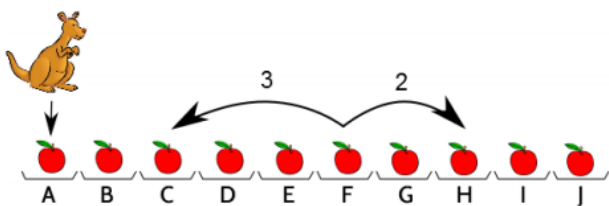


Figura 1. Canguro

Tomás sólo salta hacia platos con manzana. Cuando salta, come la manzana. *Pregunta:* Si Tomás recoge todas las 10 manzanas, ¿cuál manzana recoge al final? B, C, D, E, F, G, H, I, o J.

*Respuesta correcta:* I

2.2. **Castores en movimiento.** Una colonia de castores está viajando a través de un bosque oscuro. El camino es estrecho, así que tienen que viajar en una fila sin pasar uno del otro. Algunas veces hay un hoyo en el camino. Un hoyo es cruzado de la siguiente manera:

- Primero saltan tantos castores como sean necesarios para llenar el hoyo.
- La colonia entera pasará entonces a través del hoyo.
- Los castores que saltaron prepararán para salir del hoyo, y unirse al final de la línea.

La Fig. 2 muestra cómo 5 castores pasan un pequeño hoyo que se llena con 3 castores.



Figura 2. 5 castores que pasan por un hoyo

Una colonia de 7 castores pasa a través del bosque. Cruzan 3 hoyos. El primer hoyo se ajusta a 4 castores, el segundo se ajusta a 2, y el último hoyo se ajusta a 3 castores, como se puede observar en la siguiente figura:

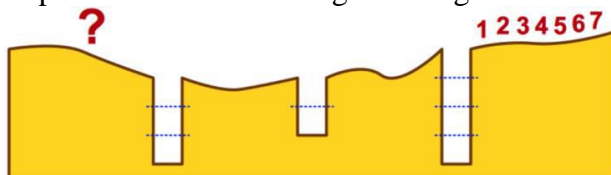


Figura 3. 7 castores cruzan 3 hoyos

*Pregunta:* ¿En qué orden se encontrarán los castores después de que hayan pasado el tercer hoyo?

*Respuesta correcta:* 2 1 6 5 3 4 7

2.3. **Espías.** Cada viernes, seis espías intercambian toda la información que han reunido durante la semana. Un espía nunca puede ser visto con más de otro espía al mismo tiempo. Así, tienen que tener varios encuentros de reunión en pares y compartir la información que poseen. El grupo de 6 espías sólo necesitan 3 encuentros para distribuir todos sus secretos. Antes del encuentro cada espía mantiene una sola pieza de información (espía 1 conoce ‘a’, espía 2 conoce ‘b’, etc.). En el primer encuentro espía 1 y 2 se encuentran y comparten información entonces ahora ambos conocen ‘ab’. La Fig. 4 muestra cuales espías se encuentran en cada reunión a través de una línea. También muestra cuáles piezas de información tienen todos.



Después de 3 encuentros toda la información ha sido distribuida.

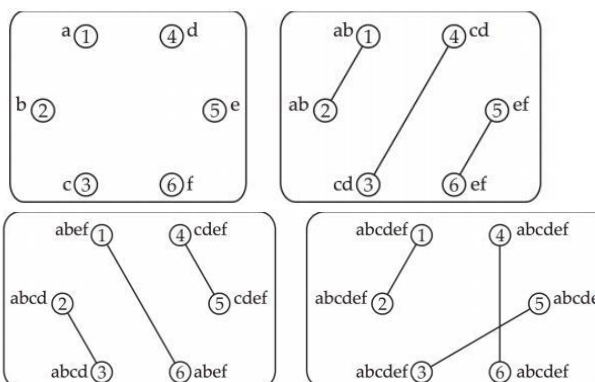


Figura 4. 1°, 2° y 3° encuentro de espías

*Pregunta:* Después de un incidente internacional un espía ha dejado de atender los encuentros. ¿Cuál es el número mínimo de reuniones necesitadas por los cinco espías restantes para intercambiar toda la información?

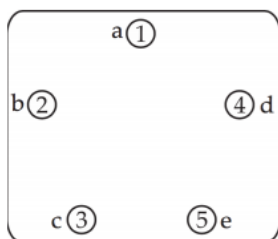


Figura 5. encuentro de 5 espías

*Respuesta correcta:* 4

2.4. **Salto de charcos.** Ana (edad 7), Berta (edad 8), Carlos (edad 9), Dora (edad 10) y Luisa (edad 11) están jugando un juego donde saltan de un charco a otro. Ellos han ubicado flechas entre los charcos, y todos inician del lado izquierdo como se indica en la Fig. 6.

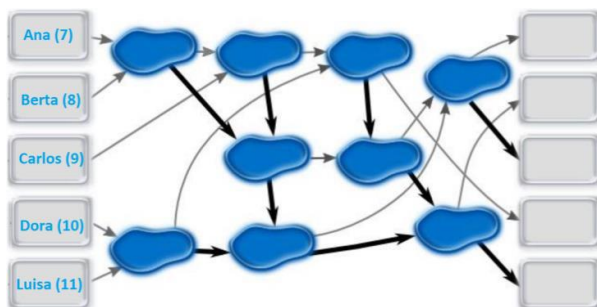


Figura 6. Salto de charcos

Cuando un niño salta dentro de un charco, espera la llegada de un segundo niño. Seguidamente el niño mayor en el charco saltará de acuerdo con la flecha gruesa, el más joven sigue la flecha delgada.

*Pregunta:* ¿Cuál es el orden (de arriba hacia abajo) en el cual los niños terminarán a la derecha?

*Respuesta correcta:* Berta, Dora, Carlos, Ana y Luisa (8-10-9-7-11)

2.5. **Móviles.** Un móvil es una pieza de arte que cuelga del techo, generalmente en los dormitorios. Un móvil consiste de varillas y figuras. Cada varilla tiene unos cuantos puntos donde figuras u otras varillas pueden ser atados. Además, cada varilla tiene un punto para colgar, donde se cuelga a una varilla hacia abajo (o hacia el techo). Un móvil puede ser descrito usando números y paréntesis.

Ejemplo: La expresión  $(-3 (-1 1) (1 1)) (2 3)$  describe el móvil de la Fig. 7.

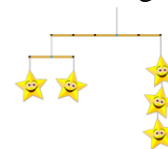


Figura 7. Móvil ejemplo

*Pregunta:* ¿Cuál de los móviles de la Fig. 8 puede ser construido usando las siguientes instrucciones?  $(-3 (-1 4) (2 (-1 1) (1 1))) (2 (-1 6) (2 3))$

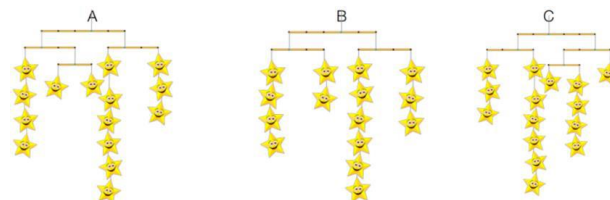


Figura 8. Móviles ejemplos

*Respuesta correcta:* A

Los ejercicios planteados de la pregunta 5 a la 9 fueron extraídos de la competencia “Desafío del pensamiento computacional 2015” de Bebras (<http://www.bebas.org>) y la Olimpiada de computación de Búsqueda de talento 2015 (<http://www.olympiad.org.za>).

## Resultados

450 estudiantes de los 700 que componen la totalidad de los que cursan la asignatura en el ciclo lectivo 2022 resolvieron el cuestionario. A continuación, los resultados destacados:

### 1. Composición de género

Mujeres: 26% y Varones: 74%.

Esta composición de género es propia de la carrera, en años anteriores los porcentajes fueron similares.

### 2. Experiencia previa de los estudiantes

Del total de respuestas, el 54% de los estudiantes señala que tiene experiencia previa en programación. En la Fig. 9 se puede apreciar que el 26% tiene experiencia adquirida en la escuela primaria o secundaria, un 3% en otra carrera universitaria o terciaria y un 25% indica que ha estudiado programación por su cuenta. Este último porcentaje da cuenta de la motivación de estos estudiantes y de la gran difusión que se le da actualmente a la programación, dada la fuerte demanda laboral de recursos humanos con esta formación.

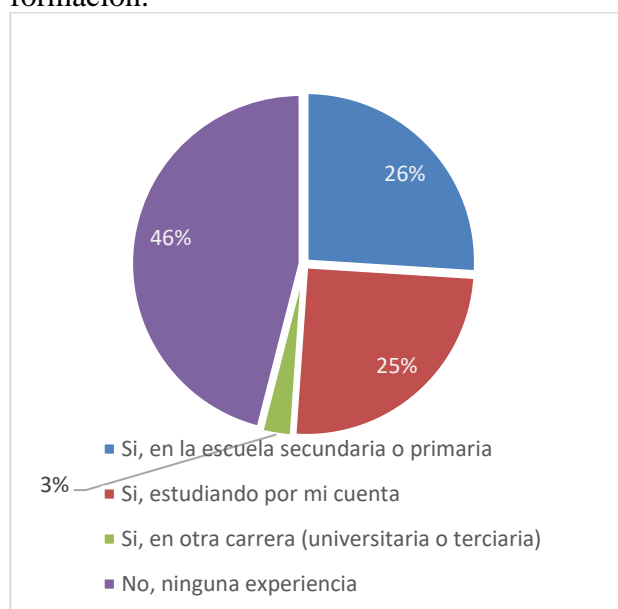


Figura 9. Experiencia en programación

A los que indicaron tener experiencia previa en programación se les pidió que señalen qué herramientas conocen. La mayoría no contestó la pregunta, pero se puede observar que los que adquirieron experiencia en la escuela primaria o secundaria señalaron que realizaron actividades

con PilasBloques, Scratch y otras herramientas lúdicas. Estas cifras dan cuenta del avance en las escuelas de la propuesta formativa que la Fundación Sadosky lleva adelante a través de su iniciativa Program.AR. Los que estudiaron por su cuenta conocen PilaBloques y lenguajes de programación convencionales (C, Java, Python, otro).

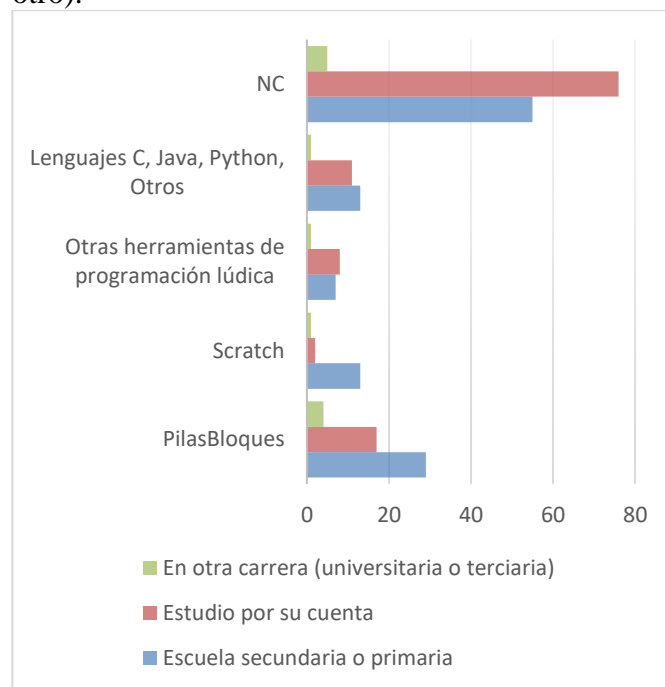


Figura 10. Herramientas de programación

### 3. Desafíos vinculados con las habilidades del pensamiento computacional

Con relación a los desafíos planteados se observan los resultados obtenidos.

#### 1. Canguro

Este desafío requiere en gran medida de la habilidad de **abstracción**, dado que el estudiante debe analizar y determinar la información relevante para lograr resolver el problema planteado. En la tabla 1 se muestra la frecuencia de las distintas respuestas. Un 45% logró resolver este desafío correctamente.

Tabla 1: Respuestas de la actividad Canguros

Respuesta	Cantidad	%
A	10	2%
B	50	11%
C	19	4%
D	14	3%
E	7	2%
F	16	4%
G	10	2%

Respuesta	Cantidad	%
H	15	3%
I	203	45%
J	97	22%
N/C	9	2%
Total general	450	100%

## 2. Castores en movimiento

Para evaluar la habilidad del **diseño algorítmico** se propuso el desafío castores en movimiento, dado que es necesario organizar las instrucciones, es decir las indicaciones para que los castores superen los obstáculos. Se obtuvo una amplia variedad de combinaciones de números como respuesta. En la tabla 2 se detallan las respuestas que se presentaron con mayor frecuencia. Un 43% logró resolver con éxito la actividad.

Tabla 2: Respuestas de la actividad Castores

Respuesta	Cantidad	%
2-1-6-5-3-4-7	192	43%
3-4-5-6-7-1-2	36	8%
7-6-5-4-3-2-1	19	4%
2-1-6-5-7-4-3	14	3%
2-1-5-6-3-4-7	11	2%
N/C	6	1%
Otras respuestas incorrectas	172	38%
Total general	450	100%

## 3. Espías

Este desafío permite evaluar la habilidad de **generalización**, o **detección de patrones**, ya que, en base a la solución indicada para la situación inicial, es posible aplicar la misma en problemas con similares características. Un 34% logró resolver este desafío correctamente.

Tabla 3: Respuestas de la actividad Espías

Respuesta	Cantidad	%
4	154	34%
3	93	21%
5	62	14%
2	37	8%
6	33	7%
Más de 7	52	12%
N/C	19	4%
Total	450	100%

## 4. Salto de charcos

El propósito de este desafío es determinar la habilidad de **evaluación**, dado que se deben

reconocer y determinar la realización de procesos, en términos de eficiencia y uso de recursos

En la tabla 4 se detallan las soluciones que se presentaron con mayor frecuencia. Un 54% de los estudiantes logró resolver este ejercicio.

Tabla 4: Respuestas de la actividad Salto de charcos

Respuesta	Cantidad	%
8-10-9-7-11	244	54%
8-9-10-7-11	17	4%
11-10-9-8-7	11	2%
9-10-8-7-11	10	2%
Combinaciones incorrectas elegidas por menos de 10 estudiantes	132	29%
N/C	36	8%
Total general	450	100%

## 5. Móviles

En el desafío de los móviles se pretende evaluar la habilidad de **descomposición**, la cual es necesaria para determinar cada componente y subcomponente del móvil. El 44% de los estudiantes resolvió correctamente este desafío.

Tabla 5: Respuestas de la actividad Móviles

Respuestas	Cantidad	%
A	199	44%
B	112	25%
C	86	19%
N/C	53	12%
Total	450	100%

## Análisis global de las actividades

Haciendo un análisis completo, en la tabla 6 se muestra el resultado de cada actividad. El porcentaje de las respuestas correctas está calculado sobre los 450 participantes.

Tabla 6: Respuestas a los desafíos planteados

Desafío	Habilidades	Correctas	%
Canguro	Abstracción	203	45%
Castores	Diseño de Algoritmo	192	43%
Espías	Generalización o detección de patrones	154	34%
Salto de charcos	Evaluación	244	54%
Móviles	Descomposición	199	44%

En promedio, el 44% de los estudiantes pudo revolver las actividades propuestas. La actividad vinculada con la habilidad de Evaluación tuvo el mayor porcentaje de respuestas correctas (54%) y el menor valor el desafío vinculado con la habilidad de Generalización o Detección de patrones (34%).

### Relación del desempeño con el sexo

En la Tabla 7 se puede ver la cantidad de respuestas correctas obtenidas entre distintos sexos, donde se observa una diferencia de media de 0.49 entre ambos grupos. Para definir si existe una diferencia entre estas categorías en la población, primero fue necesario determinar si la misma sigue una distribución normal y en función de ello elegir el test estadístico apropiado. Para ello se empleó el test de normalidad de Shapiro Wilk en la muestra reunida de respuestas correctas, y el mismo resultó significativo, rechazando la hipótesis de normalidad. Una vez determinada la distribución no normal de las respuestas correctas, se recurrió al test no paramétrico Mann-Whitney, obteniéndose un p-valor de 0.002 con un nivel de confianza de 95% demostrando una diferencia significativa entre los sexos.

Tabla 7: Estadísticas descriptivas respuestas correctas entre sexos.

Sexo	Cantidad	Media	Desvío estándar
Masculino	334	2.33	1.43
Femenino	116	1.84	1.43

### Relación del desempeño con la experiencia previa en programación

Se contabilizó para cada estudiante la cantidad de respuestas correctas, en un rango de 0 a 5, y se relacionó con la experiencia declarada por el estudiante. En la Tabla 8 se muestran los resultados.

Considerando que los que tuvieron 4 o 5 respuestas correctas presentaron un buen desempeño, estos valores se analizaron en función de la experiencia. En la tabla 9 se muestra la relación.

Tabla 8: Experiencia y desempeño

Experiencia previa	Respuestas correctas					
	0	1	2	3	4	5
Si, en la escuela secundaria o primaria	15	21	27	29	14	11
Si, estudiando por mi cuenta	12	29	23	27	17	5
Si, en otra carrera (universitaria o terciaria)	1	3	3	2	4	
No, ninguna experiencia	35	42	46	43	29	12

Tabla 9: Experiencia y mejor desempeño

Experiencia previa	Rpta. 4y5	%
Si, en la escuela secundaria o primaria	25	27%
Si, estudiando por mi cuenta	22	24%
Si, en otra carrera (universitaria o terciaria)	4	4%
No, ninguna experiencia	41	45%
Total	92	100%

Un total de 92 estudiantes (20% del total) presentaron un buen desempeño contestando correctamente 4 o 5 de los desafíos.

El dato llamativo es que el 45% dijo no tener ninguna experiencia previa, un valor que supera significativamente a los que manifestaron tener experiencia en otras carreras de nivel superior. Por lo que la experiencia, si bien se muestra como condición relevante, dado que tiene el mayor porcentaje, no está muy alejado del valor obtenido de los que no tienen experiencia previa en programación.

Profundizando en este análisis se incluyeron técnicas de estadística descriptiva e inferencial. En la Tabla 10 se presenta la cantidad de respuestas correctas obtenidas en distintos niveles de experiencia de programación. Es observable que la media de respuestas correctas mínima fue registrada por los sujetos sin experiencia (1.99) y la máxima por el grupo con experiencia universitaria o terciaria (2.90). Dada la no normalidad de las respuestas correctas, se utilizó el test no paramétrico Kruskal-Wallis. El mismo registró un p-valor de 0.001 con un nivel

de confianza de 95% demostrando una diferencia significativa entre los cuatro grupos. En cambio, al excluir el grupo con experiencia universitaria no se encontró esta misma diferencia (p-valor de 0.056 con 95% de confianza). Aquí se puede concluir que la experiencia universitaria o terciaria previa fue un motivo para obtener mejores resultados en actividades.

Tabla 10: Estadísticas descriptivas respuestas correctas entre sujetos con distinta experiencia en programación.

Experiencia previa	Cant.	Media	Desv. Estándar
Si, en la escuela secundaria o primaria	98	2.07	1.44
Si, estudiando por mi cuenta	119	2.39	1.44
Si, en otra carrera (universitaria o terciaria)	41	2.90	1.50
No, ninguna experiencia	188	1.99	1.38

## Conclusiones

Estos resultados dan cuenta de que es necesario fortalecer en los estudiantes las habilidades vinculadas con el pensamiento computacional. Surge también que la experiencia previa en programación no fue un factor determinante, excepto en el caso de la experiencia adquirida en otra carrera universitaria o terciaria.

Con lo cual, es importante adoptar metodologías de enseñanza que contribuyan a la adquisición de habilidades computacionales, mediante actividades diseñadas para tal fin.

A futuro se propone continuar con la evaluación de las habilidades del pensamiento computacional con otras actividades a fin de comparar los resultados.

## Referencias

- [1] Katz, "TIC, digitalización y políticas públicas," *Entornos Digitales y Políticas Educativas*. IPE-UNESCO, pp. 17-58. 2016.
- [2] A. Rabosto, M. Zukerfeld. "El sector argentino de software: desacoples entre empleo, salarios y educación". *Ciencia, Tecnología y Política*.

Año 2-Nº2. Enero-Junio 2019. ISSN 2618-3188.

- [3] I. Sáenz Córdoba. "Déficit del capital humano en el área informática en Costa Rica". *Technology Inside*. Vol.4-Nº 4. Agosto-Diciembre 2019: pp. 21-28. ISSN: 2215-5392.
- [4] I. Miliszewska y G. Tan. "Befriending computer programming: a proposed approach to teaching introductory programming," *Issues in Informing Science and Information Technology*, vol 4, 2007.
- [5] P. Compañ-Rosique, R. Satorre-Cuerda, F. Llorens-Largo, y R. Molina-Carmona, "Enseñando a programar: un camino directo para desarrollar el pensamiento computacional". *RED. Revista de Educación a Distancia*, vol. 46. [http://www.um.es/ead/red/46/faraon\\_et\\_al.pdf](http://www.um.es/ead/red/46/faraon_et_al.pdf). 2015.
- [6] C. Kazimoglu, M. Kiernan, L. Bacon y L. Mackinnon, "A serious game for developing computational thinking and learning introductory computer programming," *Procedia - Social and Behavioral Sciences* 47, pp. 1991–1999, 2012.
- [7] Dapozo, G. N.; Greiner, C. L.; Petris, R. H. (2017). *Introduction To Programming Based On Playful Activities In The University*. *Proceedings XLIII CLEI-46 JAIIO*, 1 (2017): 1 - 15.
- [8] Wing, J. M. (2011). *Research Notebook: Computational Thinking--What and Why?* Obtenido de *The Link*, The magazine of the Carnegie Mellon University School of Computer RED. *Revista de Educación a Distancia*. Núm. 63, Vol. 20. Artíc. 4, 30-04-2020 DOI: <http://dx.doi.org/10.6018/red.409991>
- [9] A. Csizmadia, P. Curzon, M. Dorling, S. Humphreys, T. Ng, C. Selby, J. Woollard. "Pensamiento Computacional Guía para profesores". (2015). Guía traducida al español por Codemas.org Fuente original *COMPUTING AT SCHOOL*. Disponible en: <https://www.codemas.org/wp-content/uploads/2016/04/Pensamiento-computacional-Gu%C3%ADa-para-profesores.pdf>
- [10] Selby, C. C. (2015). *Relationships:*

computational thinking, pedagogy of programming, and Bloom's Taxonomy. WiPSCE '15 Proceedings of the Workshop in Primary and Secondary Computing Education (págs. 80-87). London, United Kingdom: ACM New York, NY, USA. doi:10.1145/2818314.2818315

- [11] H. Pérez Narváez y R. Roig-Vila, “Entornos de programación no mediados simbólicamente para el desarrollo del pensamiento computacional. Una experiencia en la formación de profesores de Informática de la Universidad Central del Ecuador”, RED-Revista de Educación a Distancia, 46(9), 2015.
- [12] Rojas-López, A, García-Peñalvo, F.J. (2020). Evaluación de habilidades del pensamiento computacional para predecir el aprendizaje y retención de estudiantes en la asignatura de programación de computadoras en educación superior. RED. Revista de Educación a Distancia. Núm. 63, Vol. 20. Artíc. 4, 30-04-2020 DOI: <http://dx.doi.org/10.6018/red.409991>, Página 38 de 39
- [13] G. Simari, “Los fundamentos computacionales como parte de las ciencias básicas en las terminales de la disciplina Informática,” VIII Congreso de Tecnología en Educación y Educación en Tecnología. ISBN 978-987-1676-04-0, 2013.

**Título:** Formación de Promotores en Seguridad Informática para el ámbito escolar

## Resumen

En este trabajo describimos una experiencia de enseñanza de contenidos de Seguridad Informática a estudiantes de nivel secundario. En tanto experiencia de extensión universitaria y para poder llegar a un gran número de estudiantes, nos propusimos formar promotores de seguridad informática que pudieran dar charlas en escuelas. Participaron del proceso de formación docentes de Ciencias de la Computación, miembros del Córdoba Technology Cluster y del Ministerio de Ciencia y Tecnología de la provincia de Córdoba. A partir de un curso de 22 hs distribuidas en tres meses, 27 jóvenes promotores de seguridad informática diseñaron y ofrecieron 19 talleres de seguridad informática en escuelas primarias y secundarias de Córdoba. La demanda de talleres superó nuestra capacidad. La comunidad educativa toda mostró mucho interés en la temática y los medios de comunicación locales se hicieron eco. Lxs promotores de seguridad informática abordaron los contenidos que habían aprendido en el curso incorporando su impronta didáctica para llegar a sus pares. Nuestras observaciones dan cuenta que estudiantes y comunidad educativa en general no tienen acceso a contenidos del área de seguridad informática, aún teniendo necesidad de conocerlo para participar del mundo digital. Por otro lado, la experiencia nos permitió construir modelos de enseñanza de conceptos de Ciencias de la Computación a través del área de seguridad informática.

Palabras clave: Seguridad informática, escuela secundaria, programación, Ciencias de la Computación

## Introducción

La enseñanza de la seguridad informática es un área de creciente interés a nivel mundial que se ha profundizado a partir de la Pandemia de COVID19.

Los ciberdelitos crecieron un 300% durante la pandemia. En Argentina, se registraron incrementos considerables de nuevos usuarios de *home banking*, *mobile banking*, pagos remotos y un incremento del comercio electrónico. Se tornaron cotidianos también los trámites remotos ante la Administración Pública Nacional a través de la plataforma de Trámites a Distancia (TAD). La cantidad de apps para dispositivos móviles inundó el panorama diario y muchas aplicaciones que ya estaban en funcionamiento incrementaron su adopción de manera muy fuerte. En este contexto, diversos reportes nacionales dan cuenta de un crecimiento exponencial de la ciberdelincuencia, proporcional al volumen de maniobras ligadas a la informática que tienen lugar. Entre las modalidades delictivas relevadas se incluye el fraude, *phishing*, acceso ilegítimo, usurpación de identidad, *ransomware*, acoso, y acceso indebido a cuentas bancarias. Un gran porcentaje de estos ataques afectaron a jóvenes a través de delitos de extorsión y acoso sexual (UFECI, 2020).

Las escuelas demandan ayuda en este tema. La prueba APRENDER 2017, arrojó que el 54% de estudiantes secundarios respondieron que entre ellos se insultan y amenazan por redes sociales. Esto es común en muchos países. Algunas encuestas muestran que si bien el 82% de los directores observan esta situación, sólo un 35% de ellos consideró que la institución contaba con herramientas para abordarla (Slipczuk, 2008).

Introducir la programación con proyectos de ciberseguridad puede acercar a muchos jóvenes a carreras informáticas. Sin embargo hay escasa formación sobre seguridad informática desde las escuelas, a pesar de que muchas de ellas usan redes sociales. Esto se liga con la casi nula enseñanza de conceptos de programación y Ciencias de la Computación en general, que son claves para entender el procesamiento de la información. Para achicar la brecha digital es necesario ofrecer a los jóvenes saberes en el área que les permitan comprender y tomar decisiones sobre cómo usan las redes y transitan su vida digital.

En Argentina muchos jóvenes y niños son víctimas del cibercrimen. Una [encuesta de la empresa de seguridad informática ESET \(2013\)](#), mostró que el 75,4% de los niños y adolescentes fue víctima de grooming. Un 49,8% de ellos a través de chats, un 23,2% a través de videojuegos, un 22,7% a través del correo electrónico y un 15,3% mediante SMS o plataforma de teléfono móvil. A nivel provincial, se puede observar que el 28% de las causas de cibercrimen provienen del grooming y la pornografía infantil. En nuestro país hay más de 24 millones de usuarios de Facebook, de los cuales 16 millones ingresan todos los días (según datos de 2015 en la ciudad de Buenos Aires). Existen alrededor de 200 casos mensuales por acosos, amenazas, hostigamiento y hasta abusos que derivan en causas judiciales que se generan en el país sólo a través de esta red social.

Una encuesta realizada de manera conjunta entre Taringa! y el Consejo de Niños, Niñas y Adolescentes de la ciudad de Buenos Aires reveló que más de las mitad de los chicos de entre 14 y 17 años que tienen un perfil en la red social se encontró con un desconocido con el que chateó en Facebook. De ese universo, sólo dos de cada diez sintieron miedo. A esto hay que sumarle que el 57% de los chicos tiene computadora o dispositivo digital en la habitación.

Sin embargo, las escuelas no reciben acompañamiento para trabajar estos temas. En parte porque los conceptos de programación y Ciencias de la Computación que permitirían comprender y prevenir no son parte de la currícula obligatoria. Un 52% de un grupo de más de 500 docentes de todo el país informaron en una encuesta que en las aulas reciben de los niños quejas o denuncias por acoso, pero solo el 16% de los docentes manifestó abordar este problema (Libres de Bullying). Solo un 3,25% de los egresados de secundaria han tenidos materias de informática (Anuario del Ministerio de Educación, 2017). Generalmente la informática queda relegada a un instrumento accesorio complementario para aprender y mostrar información. La formación del usuario de computadoras y dispositivos no es suficiente para comprender cómo se procesa nuestra información.

Asimismo, presenciamos un contexto donde se está volviendo a debatir la legitimidad de una Ley de hace 10 años como la ESI (Educación Sexual Integral). Lo que manifiesta que al menos un grupo social presiona por seguir manteniendo tabú temas como abusos sexuales que hoy se dan también por redes sociales.

En síntesis, el gran número de casos de ciber crímenes contra niños y jóvenes, la escasa formación de los docentes en el tema, la poca oferta formativa en las escuelas y una sociedad que todavía debate la legitimidad de formar en Educación Sexual, nos comprometen a trabajar en esta área con los jóvenes.



En función de lo expuesto, en esta experiencia nos propusimos formar a estudiantes que cursan algunas de las orientaciones en informática en el secundario como Promotores de Ciberseguridad para que ofrezcan talleres de seguridad informática en primaria y ciclo básico de secundaria en otras escuelas. Difundir acciones de prevención en base a ofertas que las redes provocan a los jóvenes.

## **Marco Conceptual**

En los últimos 5 años hemos visto un creciente desarrollo didáctico sobre el tema de la ciberseguridad destinado a la población de niños, niñas y jóvenes por parte de organizaciones civiles y estatales (Ej: Program.ar; Argentinacibersegura). Aunque existe mucho material en la web, es escaso el desarrollo de estos temas en las aulas. Esto yace en la escasa o nula formación de alumnos y docentes en asuntos de seguridad informática y el temor por tratar un tema que parece tabú en esta franja etaria.

Según Gutierrez (2013) parte del problema es que la mayor fracción de la actividad online se realiza en solitario y en lugares privados como las habitaciones. Este hábito impide que las familias estén informadas sobre lo que hacen los jóvenes con la computadora. Docentes y padres se encuentran inadvertidos sobre la realidad que atraviesan los adolescentes que terminan apartados, rodeados de amistades peligrosas y atados a una red que pretende controlar sus vidas.

Encontramos diferentes experiencias de formación de jóvenes en el área de ciberseguridad. Algunas de ellas han avanzado desde el estado en el desarrollo de un currículum para primaria y secundaria en seguridad informática. Los resultados de un estudio sobre la implementación del currículum en el estado de Iowa (EEUU) arrojaron que el 42% de 474 alumnos registrados en el programa superaron los 250 desafíos propuestos. De ellos, 26 estudiantes fueron becados para cursar una educación más avanzada en ciberseguridad (Paller, 2017).

El plan de estudios I-SAFE incluye cinco lecciones y actividades de empoderamiento de los jóvenes en las áreas de ciudadanía cibernética, ciberseguridad, seguridad personal, identificación de depredadores y propiedad intelectual. El currículum fue desarrollado para ser consistente con la teoría de aprendizaje constructivo de Bruner, que indica que "el aprendizaje es un proceso activo en el que los estudiantes construyen nuevas ideas o conceptos basados en su conocimiento actual / pasado" (Mishna, 2009). La intención del programa fue alentar a los estudiantes a seleccionar y transformar información, construir hipótesis y tomar decisiones. Aunque esta teoría no incluye explícitamente la seguridad en Internet, la teoría era congruente con la intención de los desarrolladores del programa de que los estudiantes elaboraran sus propias perspectivas al pensar en su comportamiento online y hablar entre ellos.

Maestros de 5to y 8vo grado ofrecieron las cinco secuencias didácticas de 40 minutos cada una de este programa durante seis semanas. Luego de la implementación los niños elaboraron conocimientos sobre: propiedad intelectual relacionado con los derechos legales de los medios comprados y la descarga ilegal de medios; seguridad en Internet relacionados con elementos tales como salas de chat, depredadores, virus informáticos y plagio; gestión del riesgo

relacionado con las percepciones de que alguien que el alumno conoce en línea intente contactarlos o dañarlos; identificación de depredadores a través de percepciones de que alguien pueda tratar de contactarlos aparentando ser un niño de su edad; buenas prácticas para compartir información personal como el nombre del alumno y el lugar en el cual se junta con los amigos; y detección de comportamiento inapropiado en línea, sitios web inadecuados, imágenes inapropiadas y riesgos de compartir la contraseña. Asimismo se tomaron pruebas pre y post test sobre conceptos centrales de ciberseguridad. Los resultados mostraron que un 88% de los estudiantes mejoró sus saberes luego de tomar estas clases.

Empoderar a los alumnos es clave para que adquieran mecanismos de defensa en la web, pero además para que reconozcan cuáles son las consecuencias de sus actos en las redes, sabiendo que muchas veces ejercen acciones que pueden ser ofensivas para otras personas con las cuales interactúan.

La formación de promotores jóvenes para que trabajen con otros jóvenes y niños plantea un camino de acercamiento, que sitúa y contextualiza el conocimiento a la realidad cotidiana del estudiante (Fernandez, 2014).

### **Descripción de la Experiencia**

Este proyecto de extensión, buscó capitalizar y profundizar el trabajo de un equipo de investigación y extensión universitaria en el campo de la enseñanza de las Ciencias de la Computación. Por ello, recuperamos un capítulo sobre la Ciudadanía Digital y la Seguridad Informática escrito por miembros del equipo para los Manuales de Enseñanza de la Ciencias de la Computación de la Fundación Sadosky.

Un objetivo central de este proyecto era formar a estudiantes que cursan alguna modalidad de orientación en informática en el secundario para que ofrezcan talleres de seguridad informática en primaria y ciclo básico de otras escuelas recuperando los contenidos del Manual. Asimismo, nos propusimos enriquecer la propuesta del manual recuperando la experiencia y mirada aguda de los jóvenes y de colaboradores del curso provenientes de la industria y la academia que son expertos en el área, quienes acompañaron a los jóvenes a planificar y a ofrecer los talleres.

Se seleccionaron escuelas secundarias con orientación en informática debido a que asumimos que en ellas circulan saberes específicos de computación. La formación en informática y la condición de jóvenes les permite a nuestros estudiantes conocer las rutinas, contextos y situaciones que atraviesan sus pares como así también, estrategias que son significativas para este grupo etario. Nos propusimos: a) construir junto con referentes del Ministerio, la Industria y los jóvenes estudiantes de secundario una selección de problemas que son actuales y preocupantes para los jóvenes de hoy en el campo de la ciberseguridad. Es decir, lejos de imponer nuestra visión sobre qué problemas de ciberseguridad son relevantes, buscamos recuperar las voces de los jóvenes para focalizar los talleres, b) establecer estrategias que permitan aprendizajes significativos entre los estudiantes que reciben los talleres. Con la participación de los jóvenes y de especialistas en el tema, buscamos seleccionar las mejores estrategias de enseñanza en los talleres para tener una llegada significativa entre los jóvenes.

Elaboramos un Programa de Formación de Promotores de Ciberseguridad destinado a Jóvenes Estudiantes de Escuelas Secundarias con orientación en Informática. Este programa preparó a estudiantes para que pudieran formar a sus pares más jóvenes en bases conceptuales de programación y seguridad informática que les permitan comprender cómo se procesa información en estas redes. Desarrollamos junto a estudiantes, de manera colaborativa, estrategias para la prevención de ataques informáticos fomentando una cultura en la que lxs jóvenes se apropien de estas tecnologías empoderando a otrxs a tomar una posición informada frente a Internet.

En el mes de Marzo de 2018 comenzaron a dictarse los talleres para lxs promotores de ciberseguridad en las instalaciones de la FAMAF. Las reuniones fueron cada 15 días aproximadamente y tuvieron una duración de 2:30hs. En total se ofrecieron 22:30 hs de clases.

Dentro del eje de la seguridad informática, se abordaron los siguientes problemas:

- Cyberbullying: uso de los medios telemáticos (Internet, telefonía móvil y videojuegos online principalmente) para ejercer el acoso psicológico entre iguales. No se trata aquí el acoso o abuso de índole estrictamente sexual ni los casos en los que personas adultas intervienen.
- Grooming: acciones deliberadas por parte de un adulto de cara a establecer lazos de amistad con un niño o niña en Internet, con el objetivo de obtener una satisfacción sexual mediante imágenes eróticas o pornográficas del menor o incluso como preparación para un encuentro sexual.
- Sexting: realización de fotografías o videos de sí misma de alto contenido erótico o pornográfico por parte de una persona y su envío por medio del celular a otra.
- Vamping: conducta de adolescentes y preadolescentes consistente en quedarse hasta altas horas de la noche, en su habitación y sin que los padres lo sepan generalmente, conectados a Internet, whatsappeando, en la red social de turno o jugando a algún videojuego.
- Ciudadanía digital.
- Movimiento de la información dentro de las redes: la nube de internet, apps de celulares, permisos otorgados a empresas.
- Abstracciones e interpretación de programas.
- Ejemplos de cómo romper una app o server a través de inputs e imágenes. Desafíos y estrategias para empoderar a los alumnos.
- Modos para una navegación más segura: navegadores, redes sociales, correo electrónico.
- Ataques a la privacidad: técnicas utilizadas por terceros para acceder a nuestra privacidad y cómo prevenirlo.

Estos contenidos están contemplados en los Núcleos de Aprendizaje Prioritarios (NAPs) de Educación Digital, Programación y Robótica. Los NAPs refieren precisamente a: *La exploración del ciberespacio en forma segura, respetuosa y responsable, procurando y preservando la identidad y la integridad de las personas, en un ámbito de socialización que facilite la construcción y la circulación de saberes.* Estos saberes se incluyen en todos los niveles del sistema educativo. No obstante, nuestro programa ofrece una selección de conceptos de

computación específicos para profundizar estos saberes.

Como parte de su formación lxs estudiantes fueron invitados a una jornada de trabajo en la empresa McAfee que en ese momento tenía una sede en Córdoba Capital. En esta jornada aprendieron los proyectos de seguridad informática que lleva a cabo la empresa y fueron socializados en los modos de trabajo de los ingenieros de software.

Durante los meses de junio y agosto, se sumaron a nuestros talleres académicos y profesionales de la industria y del Ministerio de Ciencia y Tecnología de la Provincia de Córdoba con experticia en la temática para acompañar el desarrollo de los talleres que lxs estudiantes de secundario debían preparar. Es decir hubo una primera instancia de formación hacia lxs estudiantes, y una segunda instancia en que lxs estudiantes prepararon sus propios talleres de ciberseguridad.

Además del acompañamiento de los profesores de FAMAF, recibieron el acompañamiento de un profesional del área. Estudiantes y acompañantes debían planificar sus talleres en base al enfoque de la enseñanza por problemas. Se trabajaron en clave didáctica los principales elementos de este enfoque y se les hizo entrega de una planilla didáctica (ver anexo 1) que pudiera orientar a lxs estudiantes en el desarrollo de un taller bajo el enfoque del aprendizaje basado en problemas.

Lxs estudiantes socializaron en nuestros encuentros avances parciales de sus talleres. Esto permitió compartir materiales y estrategias de enseñanza. Mientras tanto, se asignaron escuelas a los diferentes grupos para que fueran a ofrecer sus talleres acompañados de un adulto colaborador.

## **Resultados**

Para documentar los resultados de esta experiencia se recolectó información sistemáticamente sobre la asistencia de los estudiantes, se registraron situaciones de enseñanza y relatos de lxs estudiantes que daban cuenta de sus procesos de aprendizaje, y se analizaron los 13 talleres ofrecidos tanto en la “planilla” de planificación que debían completar lxs estudiantes como en la presentación que resumía el recorrido que planteaban los talleres. En base a estos datos es que reconstruimos los resultados de esta experiencia.

*Convocatoria:* del total de 30 estudiantes de nivel secundario convocados, 27 concluyeron el curso y obtuvieron su certificado. En total 9 escuelas participaron, 3 no eran de la Capital. Lxs estudiantes asistieron cada 15 días a FAMAF durante el primer semestre. Para la gran mayoría de lxs estudiantes era la primera vez que visitaban la universidad. Por ello, conversamos en nuestros encuentros sobre el boleto educativo estudiantil, las líneas de colectivos que llegan a la universidad, las becas, el régimen de cursado, la gratuidad y la oferta académica. Asimismo invitamos a profesores de la casa a saludar a los estudiantes. De esta manera pudimos acercar algunos aspectos de la vida universitaria a jóvenes del secundario.

*Desarrollo Didáctico:* La experiencia permitió desarrollar una secuencia didáctica de 9 clases que se sistematizó en un sitio web de acceso abierto:

<https://blogs.unc.edu.ar/promotoresseguridad/>. El sitio sistematiza las clases, los recursos utilizados y actividades de enseñanza. Este resultado es relevante en tanto hay pocos materiales didácticos orientados al nivel secundario que aborden el problema de la seguridad informática desde la perspectiva de los conceptos centrales de las Ciencias de la Computación. Es más frecuente encontrar sitios que abordan las conductas de los usuarios pero no explican cómo funciona la computadora y los sistemas de manera de comprender cómo mejor organizar los datos para evitar delitos.

*Formación de los docentes de informática:* Un objetivo implícito en este proyecto era poder acercar a los profesores de informática del secundario a que acompañen a sus estudiantes en temas y contenidos del área de seguridad informática. Si bien algunos profesores del nivel secundario acompañaron a su grupo de estudiantes en el aula, esto se evidenció principalmente durante los primeros encuentros. Lxs profesores estuvieron presentes en las clases los primeros encuentros y luego eligieron esperarlos fuera del aula. Se observó que ningún profesor colaboró con los talleres que diseñaron lxs estudiantes. En este punto, el proyecto podría haber anticipado un rol más explícito para lxs docentes acompañantes a los fines de ofrecer una instancia de formación docente.

*Desarrollo de redes de colaboración:* El proyecto involucró a otras instituciones con sus profesionales del área que participaron activamente. Fue notable la participación de miembros de la industria, profesores de FAMAf y profesionales del Ministerio de Ciencia y Tecnología. Con desinterés y generosidad compartían sus saberes con lxs estudiantes de secundario en clases con formato de talleres. Estos colaboradores aportaron el saber experto, y casos de estudio interesantes. La propuesta de participar fue recibida siempre de manera positiva y entusiasta. Las instituciones participantes reconocían la necesidad de la formación desde el nivel secundario en estos temas. Por lo general, estas organizaciones realizan esfuerzos de manera independiente sin reconocer los aportes de otras instituciones. En este proyecto las iniciativas de diferentes instituciones se pudieron enriquecer. Por ejemplo, la formación de los miembros del Ministerio de Ciencia y Tecnología en la legislación escolar sobre la Educación Sexual Integral (ESI) permitió que se abordaran temas de acoso sexual virtual con pertinencia y profundidad. Los profesionales de la industria compartían casos de estudio y tecnologías de vanguardia desarrolladas localmente en el área. La empresa McAfee ofreció conjuntamente con nosotros, talleres para padres y docentes sobre esta temática. Estos talleres no fueron parte del proyecto originalmente pero sí surgieron del trabajo conjunto entre las instituciones. En definitiva, el trabajo con otras instituciones permitió enriquecer la experiencia de intervención educativa. En total 18 profesionales compartieron su tiempo y saberes con estudiantes de secundario.

*Aprendizajes de lxs estudiantes:* Tres indicadores fueron tomados para dar cuenta de los aprendizajes de lxs estudiantes. 1) registros de clases que mostraron disonancia cognitiva entre saberes previos y saberes nuevos 2) las planillas de planificación de talleres que completaron lxs estudiantes en donde incluían un repertorio de conceptos trabajados en los talleres 3) la calidad de los talleres ofrecidos a sus pares en cuanto a la selección y abordaje de los contenidos de seguridad informática.

Respecto a los saberes de seguridad informática notamos que lxs estudiantes traían a la clase preguntas, preconcepciones y posiciones tomadas en base a supuestos falsos de cómo funciona la

computadora, por ejemplo, se pudo observar en algunos estudiantes concepciones erróneas acerca de la seguridad del voto electrónico, priorizando el ahorro de papel y la velocidad de procesamiento de la información por sobre el riesgo al fraude “invisible” que se puede dar en la vulneración de los votos por parte de extraños. También se evidenciaba que a muchos de los problemas con los que interactuaban a diario en la computadora, los atribuían a cuestiones “conocidas” sin investigar demasiado ni plantearse duda u objeción alguna. Es decir, si una página web anda lenta es porque el internet anda mal y no porque el navegador pudiera ralentizar ese sitio o por la cantidad de datos que almacena; si las redes o el buscador me ofrecen lo que busco, es porque ya lo busqué antes y no porque desde el celular me esté escuchando lo que hablo. Había también posiciones tomadas respecto a normas o consensos de “buen uso” de los datos. Algunxs estudiantes se resistían a las normas en nombre de la libertad de expresión. Muchas discusiones permitieron construir consensos en nuestras clases sobre la necesidad de regular las actividades que implican compartir datos personales.

Respecto a las planillas de planificación se observó que lxs estudiantes incluyeron en sus planificaciones de talleres la gran mayoría de los conceptos abordados en el curso. Sobre todo abordaron conceptos de redes y sus peligros, ciberbullying, grooming, sexting, detección de delitos y abordaje de ataques.

Finalmente en los talleres ofrecidos pudimos ver que lxs estudiantes describían con detalle los problemas de seguridad informática, analizando la dimensión técnica o computacional del problema y en función de ello, las acciones para prevenir delitos. Asimismo relataban con vocabulario específico problemas de ciberseguridad incluyendo términos como privacidad de los datos, vulnerabilidad de los sistemas, etc. La devolución que obtuvimos de lxs estudiantes fue que los talleres fueron muy bien recibidos por sus pares. Éstos prestaban atención, hacían preguntas y mostraban mucho interés. En una oportunidad un canal de televisión realizó una nota sobre los talleres. Las entrevistas a lxs estudiantes dan cuenta de cómo comprendieron la relevancia del tema y qué apropiaciones conceptuales hicieron.

### [CAPACITACIÓN CIBERPELIGROS](#)

*Extensión a la comunidad:* Como resultado de este trabajo se ofrecieron 21 talleres en 8 escuelas secundarias, 6 primarias, una iglesia evangélica y un centro para jóvenes judicializados y con discapacidad. Alrededor de 700 estudiantes recibieron los talleres de seguridad informática.

### **Conclusiones**

Un primer aprendizaje de esta experiencia es que la temática de la seguridad informática tiene mucho potencial didáctico. Es decir, pudimos elaborar una propuesta didáctica donde se introdujeron conceptos centrales de la programación, redes, sistemas operativos, criptografía, bases de datos e ingeniería del software, para analizar problemas de seguridad informática. Tradicionalmente la enseñanza de la Ciencias de la Computación se realiza con introducción a la programación. La introducción a la disciplina a través de la seguridad informática es una rica posibilidad para ser explorada, ya que permite un enfoque de deconstrucción de la máquina, de explicar cómo funciona, que es una perspectiva distinta a construir abstracciones siempre en base a programas.

Otro aspecto del potencial didáctico del área se relaciona al interés de la comunidad en general

sobre estos temas debido a que son frecuentes experiencias cercanas con delitos informáticos. Esta cercanía genera emociones en torno al tema, y la emoción guía, activa la cognición. La cercanía de los problemas de seguridad informática en la población ofrecen una enorme posibilidad pedagógica de trabajo con los contenidos. Hay jóvenes y familias deseosos de aprender para evitar delitos.

No obstante, también observamos que un conjunto de saberes que no circulan en las escuelas ni entre los usuarios de las redes sociales son necesarios para comprender cómo funcionan esas redes y cómo prevenir delitos. La alfabetización digital debería abordar este conjunto de saberes para la población general.

Observamos también que los saberes son alcanzables para los estudiantes del nivel. Los promotores podían aplicar los contextos y transferirlos a situaciones o problemáticas vividas por sus pares dando cuenta de una apropiación. Sin embargo, hubiera sido interesante poder recoger información a través de pre y post test sobre cómo fueron cambiando los supuestos sobre la seguridad informática a lo largo de las clases.

Atendiendo a las demandas de la comunidad, hubiera sido interesante poder ofrecer estos talleres a los padres y madres de las escuelas. Asimismo, para futuras ediciones se podrían pensar instancias para involucrar más a los docentes de las escuelas de manera que el saber circule institucionalmente. Quizás con un esquema de curso para docentes con estudiantes.

Finalmente, valoramos la enseñanza entre pares. Observamos beneficios tanto para los promotores de seguridad como para los estudiantes que recibían los talleres. Para los primeros, la tarea requiere rigurosidad y claridad en el tratamiento del contenido. Para los segundos, se achica la brecha generacional que es más profunda en torno a los hábitos de consumos digitales.

## Referencias

- Benotti L. y Martínez C. (Coords) (2018) Ciencias de la Computación para el Aula, Manual para docentes. Fundación Sadosky.
- ESET (2013). ¿Cuál es el panorama respecto al cyberbullying en Latinoamérica?. Blog De ESET Latinoamérica
- Fernández Alatorre, A. C. (2014). Formación ciudadana: Jóvenes y acción social. *Revista electrónica de investigación educativa*, 16(1), 29-42.
- Gutiérrez, R., Vega, L., & Rendón, A. E. (2013). Usos de la Internet y teléfono celular asociados a situaciones de riesgo de explotación sexual de adolescentes. *Salud mental*, 36(1), 41-48.
- Libres de Bullying. Recursos y Herramientas. Página libresdebullying.wordpress.com
- Mishna, F., Cook, C., Saini, M., Wu, M. J., & MacFadden, R. (2009). Interventions for children, youth, and parents to prevent and reduce cyber. *Campbell Systematic Reviews*, 2.
- Ministerio de Educación de la Provincia de Córdoba. Anuario del Departamento de Estadística Paller, 2017. Inspiring the next generation of cybersecurity experts. Sans. Cyberstar
- Slipczuk, Martín (2008) Datos oficiales sobre bullying: seis de cada diez alumnos de la secundaria identifican discriminación. Chequeado. 27 de Marzo de 2018

UFECI, Unidad Fiscal Especializada en Ciberdelincuencia en Argentina (2020) Informe Anual correspondiente al 2020.



## Anexo 1. Ejemplo de Planilla de Planificación completada por lxs estudiantes

<b>Planificación de Taller sobre Seguridad Informática</b>	
Dirigido a: (Nivel educativo) Primario (Segundo ciclo) - Secundario Título del taller: “Juntos por la Seguridad”	
<b>Problema.</b> Conductas de riesgo con uso de computadoras	Los chicos comparten información en las redes sociales sin tener conocimiento de los riesgos que le pueden suceder
<b>Objetivos/</b> Qué queremos que los estudiantes aprendan?	Que los alumnos tomen conocimientos de los distintos peligros de la publicación de información personal en redes sociales y almacenamiento en celulares
<b>Contenidos</b> Qué saberes son necesarios para lograr los objetivos?	Tipos de peligros <ul style="list-style-type: none"> <li>● Grooming</li> <li>● Sexting</li> <li>● Cyberbullying</li> <li>● Rastro digital</li> <li>● Phishing - Ransomware</li> </ul>
<b>Actividades</b> <b>1. MOTIVAR</b> y despertar el INTERÉS Introducir el problema. (5 a 10') <u>Posibilidades:</u> Autorreflexión: Ej: alguna vez a ustedes o alguien les pasó que... Video Demostración de un ataque Asociación: “cuando digo robo de identidad, ¿qué piensan?”	1. Presentación (5 minutos) 2. Preguntas disparadoras (5 minutos) ¿Qué saben sobre seguridad informática? ¿Conocen a todas las personas que los siguen en las redes? ¿Tienen celulares? ¿Qué aplicaciones conocen y/o usan? ¿Qué información comparten? ¿Saben de los riesgos de la geolocalización?
<b>2. Presentación del problema.</b> <b>Emoción</b> <u>Posibilidades:</u> artículo periodístico, caso real, una carta documento que reciban ellos, historias un personaje de la vida real, juego	Proyección del video (4 minutos) <a href="https://drive.google.com/open?id=1WIwd5utDahpt1ofu2PcXyU5z-ohgAt_9">https://drive.google.com/open?id=1WIwd5utDahpt1ofu2PcXyU5z-ohgAt_9</a>
<b>3. Recuperar saberes previos.</b> ¿Qué sabemos de estos temas? ¿Qué no sabemos? <u>Posibilidades:</u> Jerarquías: top 3 Pregunta bisagra múltiple opción. “Una clave puede ser hackeada porque...(agregar opciones).	<ul style="list-style-type: none"> <li>● Les ocurrió alguna situación similar?</li> <li>● Conocen algún caso de</li> <li>● Les hackearon su facebook / instagram?</li> <li>● Alguna persona</li> </ul>
<b>4. Actividad central</b> donde los estudiantes hacen algo con los conceptos <u>Posibilidades</u> Generar claves seguras Jerarquías: las tres conductas que NO debo hacer nunca.	Presentación en Impress (30 minutos)  Encuesta

Cuadro de ventajas y desventajas de ciertas conductas Escribir un decálogo Folletos Clase magistral interactiva	
<b>5. Cierre de la clase</b> Resumen de lo hecho. Evaluación Ticket de salida encuesta	<ol style="list-style-type: none"><li>3. Elaboración de una encuesta</li><li>4. Compartir material para futuras actividades. Proyección de capítulo de serie Black Mirror Temporada 3 Capítulo 3 “Cállate y baila”</li></ol>

## Dictado del curso “La Programación y su Didáctica 2” en FaCENA – UNNE

Ana María Company, María Cecilia Espíndola, Raquel Petris, Numa Badaracco, Gladys Dapozo

Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura

Universidad Nacional del Nordeste (UNNE), 9 de julio 1449, 3400, Corrientes, Argentina

{acompany, mcespindola, rpetris, numabadaracco, gndapozo}@exa.unne.edu.ar

### Resumen.

Se presenta la experiencia del dictado del curso “La Programación y su Didáctica 2” en FaCENA (UNNE), que se ofrece como continuación del curso La Programación y su Didáctica 1, el cual aborda con mayor profundidad los conceptos y técnicas de programación, tales como, estructuras de datos, algoritmos de ordenamiento y de búsqueda. Se presentan los contenidos, la metodología de dictado y las dificultades que aparecieron en el desarrollo del curso. Se destaca que este dictado se implementó en modalidad híbrida, accediendo a la solicitud expresa de los docentes participantes dado que esta modalidad les permitió gestionar mejor su tiempo, evitando los traslados. En cuanto a los contenidos y las estrategias de enseñanza se ofrecieron diversas actividades y herramientas que permiten a los docentes contar con recursos para transmitir estos conceptos a sus estudiantes.

**Palabras clave:** Enseñanza de la programación, didáctica de la programación, incorporación de las Ciencias de la Computación en las escuelas

### Introducción

Program.AR es una iniciativa que desde el año 2013 busca impulsar la enseñanza y el aprendizaje significativo de las Ciencias de la Computación (CC) en la escuela argentina, para cumplir este objetivo realiza varias acciones. Una de ellas orientada a la formación de docentes para mejorar capacidades para enseñar contenidos de Ciencias de la Computación en la educación obligatoria argentina. La oferta de formación abarca el dictado del curso La Programación y su Didáctica 1 y 2.

Los cursos buscan proveer herramientas para que los docentes puedan enseñar contenidos básicos de Ciencias de la Computación (específicamente, programación y algoritmia), diseñados especialmente por expertos de la Fundación Sadosky. Los cursos son correlativos y por lo tanto quienes deseen cursar el 2 deben haber aprobado el 1.

Durante el año 2020-2021, debido a la pandemia por Covid-19, los cursos se brindaron de manera virtual. Para ello las actividades se dispusieron en una plataforma de enseñanza remota que luego fue alojada en los portales de cada universidad.

En la Universidad Nacional del Nordeste, el curso La Programación y su Didáctica 1, se dictó desde el 2015 al 2017. Un total de 142 docentes, pertenecientes a los niveles educativos primario, secundario y terciario, cumplieron las condiciones de aprobación. mayoritariamente los docentes provenían del nivel secundario (Dapozo et al, 2021). Como resultado de la capacitación, los resultados permiten inferir que los docentes están altamente motivados para incorporar la programación en sus espacios curriculares, que adhieren al enfoque didáctico propuesto y que son conscientes de los beneficios de promover el pensamiento computacional en las escuelas y, en la mayoría de los casos, los docentes han cambiado su percepción sobre la enseñanza de la programación (Dapozo et al, 2018).

El curso La Programación y su Didáctica 2, está destinado a docentes de nivel primario, secundario y terciario de cualquier disciplina, que posean conocimientos básicos de programación, dando

prioridad a docentes de los espacios curriculares de Matemática, Educación Tecnológica e Informática.

El curso busca capacitar docentes en la enseñanza de las Ciencias de la Computación. Está pensado para cursantes que ya tomaron un curso inicial de programación, Se busca que los docentes adquieran conocimientos de programación relacionados a estructuras de datos con recorridos en secuencias, así como estrategias adecuadas para la enseñanza a alumnos de los niveles preuniversitarios.

Con una modalidad dinámica, en las clases se realizan actividades que les permitirán incentivar a sus alumnos a que se animen a ser creadores de programas y no sólo usuarios de aplicaciones hechas por terceros.

En esta convocatoria el dictado incorporó estrategias de hibridación, agregando a las actividades presenciales teórico-prácticas sincrónicas y también clases de consulta.

En este artículo se comenta la experiencia desde la percepción de los docentes, las dificultades detectadas y los resultados obtenidos, con la idea de realizar un feedback que permita mejorar la experiencia de enseñanza.

## Descripción de la Experiencia

El equipo docente de FaCENA UNNE que había participado en convocatorias anteriores de la iniciativa Program.Ar, se presentó a la convocatoria 2021 para el dictado de los cursos la Programación y su Didáctica 1 y 2. Habiendo sido seleccionados por la Fundación Sadosky, el equipo se abocó a la difusión del curso 1, que se dictó del 04 de septiembre al 13 de noviembre de 2021, del cual participaron 56 docentes, y cumplieron las condiciones para aprobar 29 docentes.

### *Implementación del curso*

El curso 2 se programó para ser dictado del 09 de abril al 25 de junio 2022. Se inscribieron al curso 37 docentes, de los cuales participaron 29. Muchos de los docentes provienen del interior de Corrientes, y 1 (uno) de la provincia del Chaco, tal como se observa en la Tabla 1. Además, 4 (cuatro) son docentes remotos en el “Plan Ceibal”.

Tabla 1: Procedencia de los participantes

Localidades	Cantidad
Bella Vista	5
Berón de Astrada	1
Corrientes Capital	16
El Sombrero	1
Goya	3
Itá Ibaté	1
Resistencia - Chaco	1
San Roque	1
Total	29

En cuanto al nivel educativo y área, son en su mayoría docentes del nivel secundario y se desempeñan en asignaturas vinculadas con las TIC, tal como se muestra en la Tabla 2.

Tabla 2: Docentes por nivel educativo y área de desempeño

Nivel	Física y Química	Matemática	Preceptor	TIC	Total
Primario				10	10
Secundario	1	5	1	12	19
Total	1	5	1	22	29

### *Contenidos*

Respecto de los contenidos del curso La Programación y su didáctica, se repasa y fortalece la noción de sensor y se introduce el concepto de función, aprendiendo a crear nuestros propios sensores, que ya no tienen solo la forma “Verdadero/Falso” o “Si/No” sino que pueden ser valores numéricos, texto o una estructura más compleja, como una “carta” que está compuesta por un “palo” y un “valor numérico”, entre otras opciones. Disponer de sensores que se adecuen a las necesidades del problema permite que algo complejo de leer, gracias a la creación de funciones con nombres significativos, mejore la legibilidad del código y lo deje más sencillo de interpretar.

Se profundizan los conceptos de estructuras de datos, tipos de datos, manejo de secuencias, tuplas. Se introduce el concepto de “listas” y se abordan algoritmos simples de ordenamiento, búsqueda secuencial y búsqueda binaria, con énfasis en la resolución de problemas y la optimización de las soluciones.

Se trabaja también la noción de “evento” explotando las funcionalidades más utilizadas en los smartphones permitiendo a los cursantes observar la estrategia planteada en un desafío con entorno cambiante (actividad Bowloco), y el uso de variables de una forma más completa dentro del marco creación de juegos (tipos, variables globales, locales, variables utilizadas para control de puntos, para representación de información dinámica al usuario).

Se aportan metodologías para la corrección de errores del programa.

### *Metodología*

Al inicio del curso, los participantes completaron un cuestionario de saberes previos, con el fin de identificar las ideas que tenían los cursantes acerca de qué es un dato, qué puede hacer un programa, cómo piensan que funciona un buscador como Google, entre otras. Esta actividad propició el primer debate sobre conceptos vinculados al curso. Al finalizar el cursado, se les requirió completar el mismo cuestionario, a fin de visibilizar los aprendizajes logrados.

La estrategia didáctica estuvo basada principalmente en el aprendizaje basado en problemas. Los problemas se presentaron en forma de desafíos en diferentes plataformas tecnológicas.

Para la resolución de cada desafío, el curso propone una metodología para resolución de problemas que consiste en:

- 1) Idear una estrategia de solución, y explicitarla;
- 2) Expresar la estrategia mediante alguna división en subtareas;
- 3) Identificar las partes que componen la estrategia;
- 4) Declarar y nombrar adecuadamente un procedimiento principal que exprese la estrategia ideada; y
- 5) Declarar y nombrar adecuadamente procedimientos que expresen las subtareas.

Para el desarrollo de las actividades se utilizaron los entornos: PilasBloques, Gobstones JR y AppInventor. También se presentaron actividades “desenchufadas” que no requieren el uso de dispositivos electrónicos, sino que pueden trabajarse con otros recursos como papel y lápiz u otros elementos y que permiten incorporar conceptos relacionados a programación desde una perspectiva “off-line”. En el desarrollo de estas actividades se propicia e incentiva la reflexión crítica y el trabajo colaborativo.

El curso estaba previsto ser dictado en modalidad presencial, pero debido a la demanda de gran parte de los interesados que solicitaban que fuera virtual, se acordó el dictado en modalidad híbrida, combinando clases presenciales con clases virtuales sincrónicas, brindando además un espacio adicional de tutoría virtual. Esta propuesta posibilitó la permanencia de los asistentes en el curso.

Para las instancias virtuales sincrónicas se utilizó la plataforma Google Meet. Las actividades asincrónicas se realizaron en la plataforma UNNE Virtual de la universidad, en la cual se alojó el curso provisto por Program.AR para este propósito. El equipo docente local se encargó del mantenimiento del sitio, y se agregaron recursos adicionales que se consideraron necesarios.

El espacio de tutorías fue aprovechado por los participantes para recuperar contenidos de clases a las que no pudieron asistir, realizar consultas de las actividades planteadas en clases o eliminar dudas que se les presentaron a la hora de resolver las actividades en su domicilio.

La planificación comprendía alternar una clase presencial con una clase virtual sincrónica, y se respetó la planificación de las clases estipulada. No se seleccionaron ejercicios según modalidad, sino que se siguió la planificación de contenidos según el cronograma de actividades brindado. Los únicos cambios que se hicieron a la planificación de contenidos fueron en función del tiempo que necesitaban los docentes para comprender los conceptos.

### *Algunas dificultades detectadas*

Si bien al inicio del curso se contó con una planificación detallada de cada clase, basada en la planificación proporcionada por los organizadores del curso, en varias de ellas no fue posible completar todas las actividades planificadas. Se evidenciaron dificultades en la comprensión del uso de funciones, también en la identificación de qué dato (y tipo de dato) es posible completar en los procedimientos parametrizados. Estas dificultades fueron un indicador de que era necesario utilizar más tiempo para esos temas, lo que generó un desfase respecto de la planificación original. Debido a esta situación, no se alcanzó a desarrollar completamente el tema Listas, por ejemplo. Comparativamente con el dictado de la experiencia anterior, el equipo docente ha observado que en la primera edición se alcanzó a dar más contenidos, lo que no parece tener que ver con la modalidad.

También se detectaron dificultades en la comprensión de los ejercicios de Gobstones, basados en el contexto del juego del truco. El problema parece radicar en la comprensión del dominio del problema, en este caso, el juego del truco. Si bien los enunciados están perfectamente claros, pareciera ser que el desconocimiento de las reglas del juego genera un distractor que no permite mantener el foco en la resolución del desafío y el uso de los conceptos involucrados. La idea de estas actividades es interesante, y el equipo docente entiende que el objetivo es claro en el diseño de la actividad, pero en la bajada al curso se percibe que ese objetivo no está tan claro para los cursantes y esta misma percepción se mantuvo en los dos dictados del curso. Una opción para resolver este inconveniente sería agregar desafíos de la misma complejidad, pero con otro dominio del problema

### *Resultados*

Los docentes que participaron del curso tuvieron una asistencia a las distintas actividades del 90 al 100%, lo que refleja el interés de los cursantes por la propuesta de formación.

El grupo se muestra consolidado debido a la interacción que permiten estos espacios para trabajar colaborativamente, compartir soluciones, intercambiar visiones de otras posibles soluciones a los problemas propuestos. Estos espacios son enriquecedores y amplificadores de alternativas de soluciones que se pueden bajar a las aulas y llevar experiencias que optimicen el rol docente.

Muchos docentes que participaron de esta propuesta se encuentran actualmente dando clases en sus escuelas abordando los contenidos tanto de este curso como el de la primera edición.

## **Conclusiones**

La experiencia de este año, en su segunda edición, es novedosa por su implementación en modalidad híbrida, lo cual, sumado a la posibilidad de tutorías virtuales para reforzar los conceptos, ha

favorecido la permanencia de los docentes participantes, manteniéndolos motivados y participativos a través de los diferentes canales de comunicación que se manejan en esta modalidad. El equipo docente considera como un aspecto positivo, que la modalidad híbrida logró mantener y recuperar a algunos cursantes que estuvieron por abandonar debido a la modalidad 100% presencial, y permitió que docentes de otras localidades puedan acceder al curso. Lo que lleva a la conclusión de que es factible desarrollar esta capacitación en esta modalidad y que es necesario seguir por este camino para llegar a más docentes.

Otro aspecto para destacar, que esta experiencia ha puesto de manifiesto, es la importancia de la característica de conexión off-line que poseen los entornos utilizados en el curso, que favorece una mayor aceptación por parte de los docentes participantes. De sus comentarios se deduce que una dificultad que tienen para la bajada de estos contenidos en su práctica educativa es la escasez de recursos en las escuelas, algunas con falta de equipamientos otras con falta de conexión a Internet.

Por otra parte, se observa como aspecto que requiere una revisión, es la alta carga horaria del curso que se prolonga a lo largo de tres meses, lo que requiere de un gran esfuerzo por parte de los cursantes que suman tiempo extra a las actividades normales que tienen durante la semana. La sugerencia del equipo docente sería la de reducir los contenidos o desdoblar el curso en dos partes. También acortar la duración para que no sea tan extenso, teniendo en cuenta la disponibilidad de los docentes cursantes.

También se ha observado que, aun utilizando los medios de comunicación habituales para estas convocatorias, la cantidad de interesados en estas propuestas formativas ha disminuido. Esta situación podría estar relacionada con la aparición de otras ofertas de capacitación en temas de programación y robótica educativa, diplomaturas o cursos, principalmente en el nivel terciario.

## Referencias

Dapozo, G., Greiner, C., Petris, R. (2021). "Acciones en la UNNE para promover el pensamiento computacional mediante la enseñanza de la programación"- En "Pensamiento Computacional, Programación Creativa y Ciencias de la Computación para la Educación: Reflexiones y experiencias desde América Latina". Quito. Ecuador. ISBN: 978-9978-55-203-2. Ediciones CIESPAL, 2021. PP 121-141. Disponible en: <https://ediciones.ciespal.org/index.php/ediciones/catalog/view/30/34/230-1>

Dapozo, G., Petris, R. Greiner, C., Company, A., Espíndola, C. "Formación docente para incorporar la programación en las escuelas". Actas de las JENUI, Vol. 3. 2018. PP 31-38 Disponible en: [https://aenui.org/actas/pdf/JENUI\\_2018\\_009.pdf](https://aenui.org/actas/pdf/JENUI_2018_009.pdf)

## Taller de Robótica Educativa en Nivel Inicial: una experiencia de formación docente

Cenich, Gabriela\*; Felice, Laura\*\*; Miranda, Andrea\*; Leonardi, María C. \*\* y

Mauco, María V. \*\*

INTIA (Instituto de Investigación en Tecnología Informática Avanzada), Facultad Cs. Exactas, UNCPBA. (laufelice, leonardi.carmen, virginiamauco)@gmail.com

ECienTec (Educación en Ciencias con Tecnologías), Facultad Cs. Exactas, UNCPBA. (gcenich, andrea.amiranda)@gmail.com

### Resumen.

Este artículo describe y analiza una experiencia de formación docente en Pensamiento Computacional y Robótica para maestros de nivel inicial realizada de manera virtual. La propuesta del “Taller de Robótica Educativa en Nivel Inicial” surge desde un equipo de docentes-investigadores como respuesta al requerimiento de formación realizado por el Jardín de Infantes Dr. Osvaldo Zarini de la ciudad de Tandil. El diseño del taller plantea una metodología de trabajo colaborativa entre todos los participantes a través de la propuesta de cuatro encuentros virtuales. La actividad de formación tiene como objetivo principal promover en los docentes de este nivel educativo la elaboración y puesta en aula de actividades que introducen al Pensamiento Computacional a través de la robótica educativa. El desarrollo del taller permitió a los docentes establecer relaciones significativas entre contenidos del Pensamiento Computacional y otros contenidos específicos del nivel que se evidenció en la elaboración de sus propuestas finales. En este artículo se presenta la propuesta de formación, se describe la experiencia llevada a cabo en el año 2021 y finalmente, se realizan aportes para futuros trabajos acerca de Pensamiento Computacional y Robótica Educativa en el nivel inicial.

**Palabras clave:** Pensamiento Computacional, Robótica Educativa, Nivel Inicial, Enseñanza, Formación Docente.

### 1. Introducción

En las últimas décadas, se ha abierto el debate en muchos países del mundo -como Inglaterra, Israel, Estados Unidos o Nueva Zelanda- acerca de la enseñanza de contenidos de las Ciencias de la Computación (CC) y de su inclusión en los diseños curriculares de los diferentes niveles del sistema educativo (Wing, 2010; Riesco et al., 2014). En Argentina, también hace varios años que se ha planteado la importancia de enseñar CC y el desarrollo del Pensamiento Computacional (PC) en la escuela (Fundación Sadosky, 2013). Los contenidos de las CC fueron integrados de manera transversal en todos los niveles, a partir de los Núcleos de Aprendizajes Prioritarios para la Educación Digital, Programación y Robótica, que fueron aprobados por el Consejo Federal de Educación (CFE,



2018). En 2019 se presentó el Diseño Curricular para la Educación Inicial, incluyendo dentro de las áreas de enseñanza a la educación digital, la programación y la robótica. La inclusión de estos saberes está justificada en el documento del diseño curricular ya que potencia al ser social y creativo de los niños, propiciando la participación, el trabajo colaborativo, el pensamiento crítico y la resolución de situaciones problemáticas. Por esta razón, la educación digital se aborda en este nivel como un eje transversal, y así como también los aspectos inherentes al PC, programación y robótica (DGCEP Bs.As, 2019).

Para avanzar en la integración de estos saberes en el aula del nivel de educación inicial se hace necesaria la formación docente. En este sentido, el desarrollo profesional docente según Sanjurjo (2014, p. 38) involucra “la idea de que la formación en una práctica no es un proceso en el cual sólo se recibe desde afuera una capacitación, sino que supone la implicación directa del que se forma”. En esta dirección se diseña un taller de formación docente en PC y robótica para nivel inicial que contempla el trabajo en colaboración de los maestros del jardín junto con los integrantes del equipo formador. Este taller se caracteriza por plantear actividades que permitan a los docentes construir conocimientos acerca del PC y la robótica en el marco de elaboración de propuestas de aula situadas en su propia práctica. Así se hace necesario establecer algunos conceptos en referencia al PC y la enseñanza que permitan interpretar de manera integral la experiencia desarrollada.

En relación con el PC, Wing (2010, p. 1) lo define como “el proceso de pensamiento involucrado en la formulación de problemas y las respectivas soluciones, para que se puedan representar y resolver por un agente de procesamiento de información”. Esta autora caracteriza al PC desde cuatro ejes principales: descomposición de problemas (dividir un problema en problemas más pequeños y manejables), reconocimiento de patrones (reconocer patrones en los problemas para tratar de resolverlos de forma similar a otros resueltos anteriormente), realización de abstracciones (abstraer la información para omitir la que es irrelevante a fin de resolver el problema), diseño de algoritmos (diseñar pasos que permitirán resolver el problema) (Wing, 2006).

La integración del PC en actividades del nivel inicial a través de la robótica desafía a los docentes a articular diferentes áreas del curriculum en propuestas de enseñanza innovadoras. Estas propuestas pueden ser estudiadas a partir de la noción de enseñanza propuesta por Gimeno Sacristán (citado en Trillo Alonso, 2016, p.p. 58-59) que la define como “la organización de las tareas en las condiciones más favorables para que puedan desarrollarse los procesos de aprendizaje adecuados en orden a lograr la apropiación de las posibilidades de los contenidos de un determinado currículum”. A partir de esta definición Trillo Alonso (2016) identifica algunos puntos claves para llevar a cabo la enseñanza: relevancia de los contenidos a enseñar (seleccionar contenidos significantes para despertar el interés por el aprendizaje en los estudiantes), tipos de aprendizaje que se desea promover (reflexionar acerca de los aprendizajes para tomar decisiones que permitan ajustar y mejorar la enseñanza), condiciones más favorables (identificar aquellas condiciones propicias para la enseñanza), crear un clima de aula apropiado (favorecer el interés de los estudiantes por participar y colaborar), la organización de las tareas (considerar el tipo de tareas, las previsiones y qué ayudas brindará a los estudiantes), evaluar la experiencia en el aula (monitorear los procesos de aprendizaje para poder tomar decisiones y ajustar las prácticas de enseñanza).

En este trabajo se describe y analiza la propuesta de formación “Taller de Robótica Educativa en Nivel Inicial” (RCA 193/21) destinada a docentes del Jardín de Infantes Dr. Osvaldo Zarini de la ciudad de Tandil. La actividad de formación tiene como objetivo principal promover en los docentes

de este nivel educativo la elaboración y puesta en aula de actividades que introducen al PC a través de la robótica educativa.

## 2. Descripción de la Experiencia

### 2.1 Contexto

El Jardín de Infantes "Dr. Osvaldo Zarini" es una institución preuniversitaria de nivel inicial, que depende de la Secretaría Académica de Rectorado de la UNICEN. Esta institución, al igual que otras instituciones de nivel inicial de la Provincia de Buenos Aires, recibieron en 2019 un Aula Digital (Aprender Conectados 2019), en la que se incluían, entre otros recursos tecnológicos, robots educativos ("Robotita"), tarjetas con las instrucciones que puede ejecutar el robot y tabletas para programarlos, así como también los cuadernillos "Colección de Actividades Aprender Conectados. Nivel Inicial. Robótica"<sup>1</sup>.

La mayoría de los docentes del jardín no tenían formación en temas relacionados a la programación y la robótica pero querían comenzar a trabajar en estos temas con los nuevos recursos, en línea con la propuesta del nuevo diseño curricular, con una mirada crítica y adaptada a las características de su institución. Con este fin se contactaron con la Secretaría de Extensión de la Facultad de Ciencias Exactas, la que convocó a un grupo de docentes-investigadores para que realizaran una propuesta de taller para este jardín. El grupo de trabajo se conformó con docentes de dos núcleos de investigación INTIA (Instituto de Investigación en Tecnología Informática Avanzada) y ECienTec (Educación en Ciencias con Tecnologías), ambos de la Facultad de Cs. Exactas, y una docente-investigadora de la Facultad de Cs. Humanas. Diecisiete integrantes del equipo docente del jardín participarían del taller, incluyendo los directivos, los maestros de las distintas salas (3, 4 y 5 años), preceptores y maestros de Música, Inglés y Educación Física.

### 2.2 Diseño del Taller de Formación

A fines de 2019 se iniciaron las reuniones con las partes interesadas para establecer objetivos de las actividades según los intereses del jardín y las propuestas de sus docentes. Se diseñó a través del trabajo conjunto de docentes de nivel inicial con los docentes-investigadores un plan de encuentros presenciales para llevar a cabo en 2020 que incluía, en principio, actividades con los maestros y posteriormente el trabajo con los niños en el aula. Sin embargo, la situación mundial como consecuencia de la pandemia por Covid-19 requirió adaptar la propuesta inicial del taller a la modalidad virtual para poder dar continuidad al trabajo iniciado el año anterior. De esta manera, se diseñó un Aula Moodle para dar soporte a la propuesta de formación que incluyó links para posibilitar los cuatro encuentros sincrónicos, foros para intercambios y consultas, materiales, recursos y bibliografía. En el taller se plantea una metodología de trabajo colaborativa (Ingram y Hathorn, 2004) entre todos los participantes a través de la propuesta de cuatro encuentros virtuales que se describen a continuación.

El *primer encuentro* tiene como finalidad introducir a los docentes en la construcción de PC a través del desarrollo de sus ejes principales (descomposición de problemas, reconocimiento de patrones, realización de abstracciones y diseño de algoritmos). La primera parte tiene como objetivo establecer

---

<sup>1</sup> (<https://www.educ.ar/recursos/150309/coleccion-de-actividades-aprender-conectados-para-nivel-inicial-en-educacion-digital-programacion-y-robotica>).

la diferencia entre las TICs y las CC. Mediante una ejercitación sencilla se introducen conceptos esenciales de la programación. Se trabaja sobre los conceptos de algoritmos, instrucciones, programas con actividades sin computadora. Para ello, se presenta la “Máquina de dibujar” y se proponen actividades para resolver. Finalizando el encuentro, se introduce el entorno de programación Pilas Bloques, para favorecer la comprensión del concepto de programa, con las actividades que propone el entorno en Desafíos de Primer Ciclo (Pilas Bloques, Desafíos de Primer Ciclo)<sup>2</sup>.

La primera parte del segundo encuentro se centra en uno de los ejes del PC, el reconocimiento de patrones. En un principio se propone reflexionar acerca del significado de este concepto mediante ejemplos sencillos, con especial interés en la detección de patrones. Luego se utiliza esta idea para abordar el concepto de la estructura repetición utilizando las actividades que propone Pilas Bloques Primer Ciclo. La segunda parte del encuentro se centra en la expresión de alternativas condicionales, partiendo de situaciones que involucran actividades simples de la vida cotidiana. Se da cierre al encuentro recuperando la idea principal de las estructuras básicas para la programación en los lenguajes de programación. Se invita a seguir explorando de acuerdo a los intereses de cada docente en los distintos capítulos del entorno Pilas Bloques Primer Ciclo.

El *tercer encuentro* tiene como finalidad promover la elaboración en conjunto de actividades que promuevan el PC a través de Robotita. En la primera etapa, se realiza una revisión de los temas propuestos en los encuentros anteriores sobre PC y programación. En la segunda parte, se analizan secuencias didácticas con Robotita (pertenecientes a “Colección de Actividades Aprender Conectados. Nivel Inicial. Robótica”) con el objetivo de reflexionar y debatir acerca de los elementos que las componen y sobre los alcances y las limitaciones que su puesta en el aula implicaría. En la tercera parte, se propone a los docentes realizar por grupos una actividad integradora para el próximo encuentro. La tarea consiste en la elaboración de una secuencia didáctica para implementar en el aula que integre los conceptos de PC abordados en este taller.

En el *cuarto encuentro* se propone que los grupos de docentes presenten sus actividades integradoras para compartir y debatir acerca de nociones conceptuales referidas al PC, la articulación con otros contenidos, los requerimientos necesarios para poder implementar las propuestas y las experiencias de desarrollo en el aula.

### **3. Desarrollo y análisis de la experiencia**

A continuación, se describe el desarrollo de los encuentros de formación. De acuerdo a lo expresado en el diseño, en el análisis de los dos primeros encuentros se pone el foco en la construcción de conocimientos acerca del PC para abordar tareas que involucran la robótica educativa y en los dos últimos encuentros el análisis se centra en la integración de los conceptos de PC y robótica educativa en propuestas significativas de enseñanza.

#### **3.1 Acerca del primer encuentro**

El primer encuentro comenzó con la presentación de los objetivos del taller y la dinámica de trabajo. Al abordar los aspectos fundamentales del PC se propusieron actividades desconectadas con la

---

<sup>2</sup> Pilas Bloques. Desafíos de Primer Ciclo. <https://pilasbloques.program.ar>

“Máquina de dibujar”. Por ejemplo, pintar casillas sobre una cuadrícula de 4 x 4 partiendo de la casilla donde se encuentra la estrella. Al presentar dicha actividad surgió espontáneamente por parte de los docentes, la siguiente pregunta: ¿cómo conoce la máquina de dibujar los movimientos que ella puede hacer?. Esta pregunta fue un disparador importante para la introducción del concepto de autómatas y qué es lo que sabe hacer un autómata. Los docentes observaron que las instrucciones estaban muy claras y construir *una secuencia de instrucciones* para lograr el objetivo final, no tuvo variantes de solución. En la solución al problema de dibujar, se expresa qué significa la estrella en un casillero, entonces se abre el debate con la pregunta ¿qué pasa si la estrella (inicio) está en otro casillero? ¿es diferente la solución?. Esto permitió abordar el concepto de *condiciones iniciales de un programa*. Se trabajó el concepto de *precisión de los lenguajes de programación*, qué significan cada una de las instrucciones, distinguiendo el lenguaje natural (la ambigüedad que lo caracteriza) del lenguaje de programación y la precisión de cada instrucción. En este punto, se introdujo un panorama general del entorno de programación Pilas Bloques. Se dieron los detalles necesarios del entorno para empezar a elaborar programas y se presentó el problema Desafío 1 del Capítulo 3 correspondiente a Desafíos del Primer Ciclo.

Se seleccionaron estos desafíos dada su simplicidad, y de esta manera poder elaborar programas sin dificultades. Una primera solución al problema surgió rápidamente. En este contexto, surgió por parte de una docente la propuesta de guardar distintas soluciones para luego ser comparadas en términos de *claridad del programa* y *eficiencia en cantidad de instrucciones*. A partir de esta inquietud, se trabajó en la idea de ir creando para una misma actividad distintos desafíos para luego llevar a cabo en el aula con los niños. Así, surgió la propuesta de crear desafíos con distintos puntos de partida para cada grupo de niños a los cuales se les asigna el mismo problema.

Al trabajar con el siguiente desafío (“Coty empieza a dibujar”) una respuesta de una docente a la elaboración de la solución, hace que se establezca la diferencia entre un lenguaje coloquial y el lenguaje preciso que tienen las instrucciones. La docente dijo “saltar”, sin embargo, las instrucciones son “saltar derecha”, “saltar izquierda”, “saltar arriba”, “saltar abajo”, esto da lugar a volver a revisar con qué instrucciones cuenta el autómata y la precisión de ellas, poniendo énfasis en el concepto de ambigüedad del lenguaje coloquial o natural.

La propuesta para el cierre de este encuentro fue alentar a los docentes a agruparse para continuar con actividades similares a las trabajadas en el encuentro, actividades disponibles en el entorno de Pilas Bloques Primer Ciclo.

### 3.2 Acerca del segundo encuentro

La primera parte del segundo encuentro tuvo como principal objetivo abordar el reconocimiento de patrones dentro del PC. Detectar qué es lo que se repite para identificar un patrón fue el punto de partida. Se ejemplificó con los patrones que caracterizan a las estaciones del año, al funcionamiento de un semáforo, a las figuras que se encuentran en una serie dada. Se hizo especial hincapié en encontrar el patrón y luego, la cantidad de veces que se repite. Se trabajó sobre el capítulo 4 de Pilas Bloques Primer Ciclo con actividades de la Puma Duba. El primer desafío cuestiona: “¿Puede la puma comer el churrasco usando solo una vez el bloque mover?”. Sin esperar una respuesta por parte de los docentes, se construyó un programa utilizando una sola vez el bloque mover, observando que no se resuelve el problema. De esta manera surgió en principio, como respuesta, la *identificación del patrón* “mover a la derecha” y la utilización del bloque de la repetición. La acción “comer churrasco” fue colocada sin inconvenientes fuera del bloque repetir.

Se continuó con la actividad: ¿Cómo puede hacer la Puma para llegar a su comida sin tener que usar muchos bloques mover?, los docentes reconocieron que hay más de un patrón identificado (uno de ellos mover abajo y el otro mover a derecha), proponiendo la *secuenciación de dos bloques repetir*.

Al abordar la actividad: “La llama quiere dibujar cuatro líneas”, se preguntó: ¿qué es lo que se repite esta vez?, se identificaron sin problemas los patrones (mover a derecha, saltar), se probaron distintas cantidades y se establecieron qué instrucciones quedaban fuera del bloque repetir. El Desafío 2 de la misma actividad, presenta un pequeño agregado al final: una línea vertical a ser dibujada. La reflexión por parte de los docentes se basó en que el agregado provoca que ya no sea buena idea saltar después de trazar la cuarta línea horizontal, porque si así lo hiciera, Coty quedaría mal posicionada para dibujar la línea vertical.

La siguiente actividad: “Lita, a puro vegetal”, presentó una pequeña dificultad para encontrar el patrón. Los docentes corroboraron qué habían seleccionado el patrón correcto ejecutando la secuencia de instrucciones. Se presentó otra solución alternativa, diferenciando cuáles bloques colocar dentro y cuáles fuera de la repetición, también, en este caso, no se presentaron dificultades para abordarlos.

La segunda parte del encuentro tuvo como fin identificar el concepto computacional de la *alternativa condicional*. Para ello, nuevamente, se abordaron situaciones cotidianas como por ejemplo preguntar qué acciones simples hacemos en ciertos casos de la vida diaria: “Si entrás a tu casa y es de día, ¿prendés la luz?”; “Si vas a salir: ¿Siempre lo hacés con campera?”; Si llueve: ¿salís con paraguas?”. Es decir, hay veces que hacemos algunas cosas sólo en ciertos casos. De esta afirmación, surgió la siguiente pregunta por parte de un docente: “¿Esto corresponde en Lógica al “si y solo sí”?”, lo que dió puntapié para analizar la correspondencia de la alternativa condicional con el “entonces ( $\Rightarrow$ )” de la Lógica Proposicional.

Se trabajó sobre el capítulo 5 de Pilas Bloques Primer Ciclo en el desafío: “La puma Duba busca comer un churrasco”. Analizado el escenario del problema, se indagó: ¿En qué se diferencia este desafío de los anteriores?, la respuesta fue unánime, el escenario no es siempre el mismo: en algunas ocasiones hay un churrasco y, en otras, no. Deberíamos construir un programa que en cualquier escenario posible resuelva el problema. Se explicaron las características del bloque *condicional*. Para ejemplificar, se trabajó en el desafío en el cual Duba tiene a su derecha dos casilleros. En cada uno puede o no haber un churrasco. En un principio, los docentes identificaron que se repite dos veces un mismo patrón. Se aclaró qué si bien es adecuada la observación, en este ejercicio, no está disponible el bloque repetir para evitar dificultades con el anidamiento de estructuras. A continuación, se invitó a los docentes a seguir explorando, de acuerdo a sus intereses, en los distintos capítulos del entorno Pilas Bloques Primer Ciclo.

A modo de resumen, se pudo observar que las tareas no resultaron dificultosas y se arribaron a soluciones consensuadas. El análisis acerca de las instrucciones que proveía la actividad permitió a los docentes en conjunto realizar intercambios significativos y reflexionar acerca del PC.

### 3.3 Acerca del tercer encuentro

En el tercer encuentro se realizó una revisión de los conceptos sobre PC y programación abordados en las instancias anteriores. A continuación, se procedió a trabajar sobre secuencias didácticas que integran a Robotita (pertenecientes a “Colección de Actividades Aprender Conectados. Nivel Inicial. Robótica”) con el fin de profundizar en la comprensión de aspectos conceptuales y didácticos que su desarrollo implica. En esta instancia se promovió la reflexión acerca de las relaciones entre estas actividades y los proyectos de enseñanza que los docentes desarrollan.

La primera actividad de la colección, llamada “Una sorpresa llega al jardín” propone construir y analizar una secuencia ordenada de instrucciones. Se trata de un desafío sencillo, ir de un lugar a otro siguiendo instrucciones precisas para alcanzar una meta. Para ello la actividad propone dibujar una grilla en el piso que los niños recorren siguiendo el camino pensado por sus pares e indicado a través de las tarjetas de comandos provistas con el equipamiento recibido (Tabletas con la aplicación Blue Bot y grilla en papel). Tiene como objetivos el reconocimiento de las instrucciones representadas en las tarjetas y el desarrollo de habilidades para la orientación espacial y la acción del cuerpo en relación a las instrucciones atrás, adelante, derecha, izquierda. No se propone aún la utilización de la Robotita.

Al comienzo un grupo de docentes de sala de cinco comentó una experiencia exploratoria en la que trabajaron con la Robotita sin introducir las tarjetas de comandos. Se proponía a los alumnos que, de a uno por vez, eligieran la acción que seguiría Robotita para desplazarse por el aula y alcanzar una meta. En voz de los docentes esta exploración les permitió: “ver cómo respondía el grupo a lo que presentábamos”, “saber si estaban interesados”, “saber cómo funcionaba”, “más que armar una secuencia ordenada de instrucciones fue más exploratoria desde el uso”.

Para continuar, el diálogo se orientó a indagar acerca de la utilidad de las tarjetas de direcciones. Se buscaba problematizar a los docentes en relación con el contenido específico de PC que se ponía en juego en la actividad. Surgieron expresiones como: “hacia dónde tiene que ir el mensajero, o sea las instrucciones para hacer el trayecto”, “el algoritmo, es decir la secuencia ordenada de pasos”, “todas las tarjetas juntas serían como una especie de reglamento, de consignas”, “el lenguaje de programación”. Esto permitió a los docentes acordar significados y explicitar nociones tales como: *instrucción, algoritmo, secuencia, programa y lenguaje de programación*. También se debatió acerca de la incorporación de obstáculos para la toma de decisiones y de señales para definir los recorridos.

Finalmente, sobre el cierre de la actividad, los docentes realizaron un debate en el que manifestaron *la importancia de contenidos involucrados* como la lateralidad, la posición, el dirigir el cuerpo, el relato de las distintas posiciones, el concepto de derecha e izquierda. Ellos expresaron:

“a la secuencia le faltaría toda una primera parte para que los chicos puedan poner el cuerpo, relatar las distintas posiciones, esto de la lateralidad, alguno lo va a tener más claro que otro, ‘eso de derecha que está enfrentado a izquierda’, “derecha e izquierda es una dificultad que está incorporándose, se va adquiriendo a medida que se van trabajando distintas actividades”

La segunda secuencia didáctica de la colección, llamada “Adivina, adivinador... ¿de qué trabaja señor?”, propone realizar una grilla en el piso. En la actividad se sugiere colocar dibujos realizados por los niños referidos a oficios o profesiones en algunos casilleros. Robotita tendría que encontrar dos elementos que se relacionen con una ocupación elegida por el grupo de estudiantes. Se plantean como objetivos conocer los comandos básicos para programar a Robotita, diseñar algoritmos simples y establecer estrategias para la resolución de desafíos.

Al estudiar esta actividad en conjunto, los docentes continuaron estableciendo relaciones con la experiencia exploratoria comentada anteriormente. A través del intercambio de ideas los maestros pudieron *evaluar la experiencia realizada*. De esta manera, reconocieron *los aprendizajes que deseaban promover* tanto sobre PC: algoritmos y ejecución de instrucciones a través de los botones como algunas *condiciones más favorables para llevarlos a cabo*: ubicación del niño en referencia a la Robotita para facilitar la elaboración del algoritmo y la definición del punto de salida en la grilla. En palabras de una docente:

“Nosotras le presentamos la Robotita y le contamos que tenía botones para adelante, para atrás, para un lado y para el otro, nombramos derecha e izquierda pero que sepan que es un lado y el otro, hasta que incorporen ese concepto. Después lo que sí teníamos en cuenta que cuando ellos programaran o pusieran las opciones que se ubicaran detrás para que esté bien la orientación del recorrido.”

### **3.4 Acerca del cuarto encuentro**

De las propuestas presentadas por los docentes en el cuarto encuentro se describen a continuación, a modo de ejemplo, una secuencia didáctica para cada sección (3 años, 4 años y 5 años).

Para la primera sección (3 años) se propone realizar un paseo por la manzana donde está ubicado el Jardín. En otra etapa de la actividad se elaborará un plano con aquellos negocios e instituciones que se habían identificado en el paseo. En una próxima instancia la maestra tendrá tarjetas con las imágenes de cada residencia del plano. Entonces los niños por turno elegirán una tarjeta y desde el punto de salida (el Jardín) deberán trasladarse por la grilla hasta el destino. Luego se incorporarán las tarjetas de direcciones y una vez que se familiarizaran con ellas se propondrán secuencias de tarjetas para que los niños realicen el recorrido indicado. Por último, se incorporará la Robotita y se realizarán diferentes recorridos en el plano teniendo en cuenta la secuencia establecida a través de las tarjetas de direcciones.

En la propuesta para la segunda sección (4 años) se plantean tres actividades.

En la primera actividad se presenta a los niños una grilla de 4x4 dibujada sobre el piso. Las docentes presentan flechas realizadas en papel de dos tipos: flechas rectas para avanzar en el sentido señalado y flechas curvas para avanzar y girar. Se indica en este escenario el casillero de salida y se propone plantear posibles recorridos para llegar hasta un objeto colocado en la grilla.

En la segunda actividad se coloca otro objeto sobre la grilla y se invita a los niños a plantear distintos trayectos que partan desde la salida hasta el casillero donde se encuentra el objeto. En esta oportunidad la maestra registra las secuencias de instrucciones en la pizarra.

En la tercera actividad se invita a los niños a proponer los recorridos y que uno de ellos con la ayuda de sus compañeros escriba el trayecto en el pizarrón.

En la tercera sección (5 años) se dispone de una grilla dibujada en el piso y se presentan 2 carteles para indicar la salida y la llegada. Además, se facilitan flechas rectas para indicar el trayecto. Se propone a los niños recorrer los caminos indicados por las flechas con el propio cuerpo. Al realizar la actividad surgió la dificultad de interpretar el giro y avance representados por las flechas en el camino. Se debatió entre todos posibles soluciones y se acordó representar el giro y avance por dos flechas en el mismo casillero.

Luego, se complejizó la actividad agregando obstáculos y por último se incorporó la Robotita para realizar los recorridos planeados.

Al final, se llevó a cabo otra actividad colocando en la alfombra transparente dibujos realizados por los niños de vegetales cultivados en la huerta del jardín. En esta ocasión Robotita debió cosechar las plantas de acuerdo a los diferentes recorridos sugeridos por los niños.

En este encuentro final, al realizar la evaluación de la experiencia en el aula emergió como una dificultad relacionada con la secuenciación de instrucciones, la indicación correspondiente al giro. Esto permitió volver a trabajar sobre los conceptos de instrucción, programa, codificación y lenguaje

tratados en el encuentro anterior. De esta manera, los docentes pudieron reflexionar acerca de las implicancias de proponer símbolos (como la flecha curva) o colocar dos flechas en una casilla. Así, queda en evidencia la necesidad de que las instrucciones tengan un único significado y no presenten ambigüedades al momento de ser interpretadas para su puesta en acción.

Sobre la base de los intercambios realizados una de las docentes reflexionó acerca de la relevancia de los contenidos a enseñar y los tipos de aprendizaje que se desea promover:

“Los docentes hemos trabajado este tipo de cosas pensando los recorridos solamente por la noción de espacios entonces ahora podemos empezar desde ese lugar porque el niño está construyendo la noción de espacio pero con el propósito en términos de programación y es ahí donde esto se comienza a complejizar (...) va a ser muy rico cuando los niños lleguen a sala de 5 con todo este trabajo previo realizado.”

## **Comentarios Finales**

El abordaje colaborativo que propone el taller propicia la construcción de nuevos conocimientos para iniciar la integración de la robótica educativa en el nivel inicial. A lo largo de los encuentros los docentes lograron internalizar los contenidos de PC y fueron capaces de reconocer aquellos específicos que se ponían en juego en cada nueva situación que se presentó o se diseñó, vinculando otros contenidos como la lateralidad, la posición, el dirigir el cuerpo, el relato de las distintas posiciones, el concepto de derecha e izquierda, la representación de los giros, entre otros. De un total de 17 docentes inscriptos, asistieron a los encuentros virtuales del taller en promedio 15 docentes y presentaron en conjunto dos propuestas, una correspondiente al turno mañana y otra al turno tarde.

El análisis en clave didáctica de las secuencias provistas en los materiales y la puesta en común de las propuestas elaboradas por los docentes, permitieron debatir acerca de las posibles lógicas que pueden construirse en relación con el abordaje conceptual a promover en cada secuencia de aula. También, permitieron identificar aspectos centrales para pensar la organización de las tareas en el aula como: la disposición en pequeños grupos, teniendo en cuenta el tiempo de atención, y los requerimientos tanto tecnológicos como de recursos humanos necesarios para su implementación.

Además, el recorrido propuesto en el taller permitió a los docentes conocer y reflexionar sobre el desafío que propone la robótica educativa en el nivel inicial y que los interpela acerca de las articulaciones a establecer con los diferentes proyectos que desarrollan en las distintas áreas de trabajo del nivel (Expresión Corporal, Inglés, Escritura, entre otras).

En relación con el diseño del taller de formación, la adecuación de los encuentros a la modalidad virtual, permitió llevar a cabo el proyecto y lograr los objetivos propuestos. Esta situación dio lugar a considerar la posibilidad de ofrecer este taller de manera híbrida a una comunidad más amplia y llegar a más docentes del nivel.

Finalmente, este primer acercamiento a la problemática de integración del PC y la robótica educativa en el aula del nivel inicial permitió posicionar a los docentes de manera crítica y reflexiva ante estos nuevos contenidos y el material recibido, y comenzar un proceso de formación que desafía a los participantes de ambas instituciones a continuar elaborando respuestas en colaboración.



## Referencias

- Consejo Federal de Educación (2018). Resolución N° 343/18. Núcleos de Aprendizajes Prioritarios de Educación Digital, Programación y Robótica. Argentina: Ministerio de Educación.
- Dirección General de Cultura y Educación de la Provincia de Buenos Aires (2019). Diseño Curricular para la Educación Inicial. La Plata.
- Fundación Sadosky (2013). CC 2016: Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas. Buenos Aires: Fundación Sadosky.
- Ingram, A. y Hathorn L. (2004). Methods for Analyzing Collaboration in Online Communications en T. Roberts (ed), *Online Collaborative Learning: Theory and Practice* (pp. 215-241). USA: Idea Group Inc.
- Resolución Consejo Académico Facultad Cs. Exactas N°193 (2021). “Taller de Robótica Educativa en Nivel Inicial” (RCA 193/21).
- Riesco, M., Fondón, M. D., Álvarez, D., López, B., Cernuda, A. y Aquilino, J. (2014). La Informática como materia fundamental en un sistema educativo del siglo XXI. *Actas de las XX JENUI*. Oviedo, pp. 27-32.
- Sanjurjo, L. (2014). Razones que fundamentan nuestra mirada acerca de la formación en las prácticas en L. Sanjurjo (Coord.), *Los dispositivos para la formación en las prácticas profesionales* (pp. 15-44). Santa Fe: Homo Sapiens Ediciones.
- Trillo Alonso, F. y Sanjurjo, L. (2016). *Didáctica para profesores de a pie. Propuestas para comprender y mejorar la práctica*. Rosario: Homo Sapiens Ediciones.
- Wing, J.M, (2006). Computational Thinking. *Communications of the ACM*, vol. 49, no. 3, pp. 33-35.
- Wing, J. M. (2010). *Computational Thinking: What and Why?*.  
[www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf](http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf)

# Lógica, Programación y su Enseñanza: Un curso introductorio para la formación docente

Araceli N. Acosta<sup>1</sup> y Cecilia Martínez<sup>2</sup>

1. FAMAF - Universidad Nacional de Córdoba

2. FFyH, Universidad Nacional de Córdoba

**Resumen:** En esta investigación analizamos particularmente el primer módulo de una Especialización en Enseñanza de la Programación, denominado “Lógica, Programación y su Enseñanza”, el cual fue el más elegido como referencia entre los y las docentes para realizar su trabajo final de práctica de la misma carrera. Este módulo recupera y revisa ideas previas sobre la computación y su enseñanza. Además, introduce a los docentes en la lógica de la programación a través de juegos lógicos y situaciones problemáticas sin el uso de la computadora. En relación con la práctica de la enseñanza, se analiza la introducción de la programación en la escuela en tanto innovación educativa y se aborda su relación con la calidad educativa. Se abordan diferentes concepciones de pensamiento computacional.

**Palabras clave:** Formación docente, enseñanza de la programación, enseñanza de las ciencias de la computación

## Introducción

La Facultad de Matemática, Astronomía, Física y Computación de la Universidad Nacional de Córdoba (FAMAF) en forma conjunta con el Ministerio de Educación de la Provincia de Córdoba, ofrecieron a los docentes de Educación Primaria, Secundaria y Superior la *Especialización Docente de Nivel Superior en Enseñanza de la Programación en Informática* en los años 2018 y 2019 con el financiamiento de la Fundación Sadosky y del Gobierno provincial.

Se decidió un postítulo en enseñanza de la programación debido a que es la mejor manera que se conoce de introducir conceptos centrales de las Ciencias de la Computación (Denning, 1989). Los diferentes gobiernos a nivel global han acordado un concepto de alfabetización digital que además de incluir habilidades de uso de las computadoras, requiere de conceptos fundamentales del campo de las Ciencias de la Computación (Bocconi, 2016). Estos conceptos se han enseñado tradicionalmente a partir de la programación (Papert, 1993). Una sociedad alfabetizada en los lenguajes de nuestro tiempo, requiere el dominio de lenguajes de computación y de los modos de producción de tecnología digital (Busaniche, 2011).

La enseñanza de la programación en la escuela promueve herramientas para comprender las lógicas de funcionamiento de la tecnología digital. Es decir, se trata de reconocer el conjunto de decisiones que toman y los saberes que dominan quienes desarrollan el software que controla dichas tecnologías. El conocimiento de estos saberes alienta un modo de pensamiento afín a los desafíos de una sociedad atravesada por las nuevas tecnologías. En palabras de Palamidessi (2014), es un problema cuando el sistema educativo no considera a la tecnología como un objeto de reflexión y lo reduce a un simple medio.

La Especialización tenía como objetivo formar a los docentes en las bases de la programación y su didáctica. El plan de estudios (tabla 1) está organizado en proyectos acotados de programación

siguiendo los principios del aprendizaje por descubrimiento, enfoque pertinente y privilegiado para su didáctica.

Módulo	Formato curricular	Cant. Horas
<b>PRIMER AÑO</b>		
Lógica, Programación y su Enseñanza	Proyecto	40
Ensamble y Programación de un Robot	Proyecto	40
Administración y Configuración de Herramientas Informáticas	Proyecto	40
Introducción a lenguajes de programación a través de animaciones y videojuegos	Proyecto	40
Seminario de Trabajo Final I	Seminario	40
<b>SEGUNDO AÑO</b>		
Resolución de problemas de programación a partir del diseño de juegos interactivos	Proyecto	40
Introducción a la programación orientada a objetos a partir del desarrollo de un sistema de votación	Proyecto	40
Procesamiento de datos a partir de análisis de bases de datos públicas	Proyecto	40
Desarrollo de Aplicaciones Web	Proyecto	40
Seminario de Trabajo Final II	Seminario	40
<b>Carga horaria total</b>		<b>400</b>

**Tabla 1:** Plan de estudios

Los proyectos seleccionados tenían la intencionalidad de posibilitar la integración de contenidos fundamentales de la programación con los saberes pedagógicos y tecnológicos necesarios para su enseñanza. En ese sentido, se ofreció una formación que integró contenidos de la disciplina con saberes de didáctica y pedagogía.

Los contenidos disciplinares se organizaron en un currículum de formato espiralado a modo de nociones que fueron ampliando su dificultad en tres niveles (desde la familiaridad al uso y, luego, al dominio) a lo largo de la especialidad. De este modo, se esperaba que los y las docentes, en primera instancia, se familiarizaran con algunos conceptos; luego, los usaran en un programa o análisis y, finalmente, produjeran nuevos saberes a partir del dominio conceptual.

En esta investigación analizamos particularmente el Módulo 1 de esta especialización denominado “Lógica, Programación y su Enseñanza”, el cual fue el más elegido como referencia entre los y las docentes para realizar su trabajo final de práctica. Este módulo recupera y revisa ideas previas sobre la computación y su enseñanza. Además, introduce a los docentes en la lógica de la programación a través de juegos lógicos y situaciones problemáticas sin el uso de la computadora. En relación con la práctica de la enseñanza, se analiza la introducción de la programación en la escuela en tanto innovación educativa y se aborda su relación con la calidad educativa. Se abordan diferentes concepciones de pensamiento computacional. La pregunta de investigación que guía nuestra indagación es ¿Cuáles fueron los aprendizajes de programación y de didáctica de la programación en este módulo? Como preguntas secundarias interesa analizar: ¿Qué concepciones respecto a la computación cambiaron a partir de este módulo?

A continuación, presentamos cuáles fueron los aprendizajes de los docentes en este módulo y qué nociones sobre la computación y la enseñanza permitieron revisar. Los resultados de esta investigación pueden contribuir a informar el desarrollo de experiencias de formación docente en el campo.

## Fundamentación del módulo en el plan de estudios

En esta sección describimos los supuestos respecto a la elección de los contenidos del Módulo que analizamos. Se decidió ubicar este módulo en la primera instancia formativa porque buscábamos romper con la concepción de computadora como herramienta que veíamos presente en cada uno de los talleres y capacitaciones docentes que realizábamos en años anteriores a este postítulo. En efecto, el sistema educativo argentino ha instalado el paradigma de integración de las TIC como dominante en las escuelas. Esto ha sido muy provechoso porque permitió avanzar en la apropiación de tecnologías digitales para la representación de información. No obstante, desalentó la computación como objeto de estudio. Por la fuerte llegada que tuvo el paradigma integrador de las TIC en la escuela se equiparó saber computación con saber sobre TIC. Esta idea inicial era muy difícil de desarmar y reconstruir en nuestras capacitaciones. Por tanto se pensó un módulo introductorio que pudiera abordar las nociones básicas sobre qué implica enseñar programación.

Así, el objetivo del módulo era contribuir a la construcción de una mirada superadora de la computación como disciplina. Conocer el origen y los fundamentos de la computación nos permite posicionarnos de manera diferente frente a los procesos de transmisión del conocimiento disciplinar. Comprender que enseñar computación, presenta nuevos desafíos para la enseñanza de esta disciplina que superan ampliamente el abordaje de formación de usuarios u operadores de programas.

El módulo, que se desarrolló en 8 semanas, revisó brevemente la historia de la computación y el modo en que esta disciplina ha ingresado a la escuela a través de los años, también se presentó la idea de cómputo y de máquina universal y se plantearon juegos lógicos que permitieron introducir conceptos básicos de las ciencias de la computación, como el concepto de variable, secuencia, árbol, etc. El objetivo de la revisión de la disciplina en la escuela era permitir la construcción de un marco para analizar críticamente las prácticas de enseñanza de computación en la escuela. En particular analizamos cómo desde la década del 80 la computadora ha entrado a muchas escuelas y su ingreso ha sido acompañado con propuestas didácticas para abordar el estudio de este nuevo objeto. Las perspectivas y dimensiones de análisis para aproximarse a la computadora y su disciplina han evolucionado desde entonces. En algunos momentos históricos, se han superpuesto enfoques que, en apariencia, abordan diferentes dimensiones de la computación, tales como analizar el hardware (enfoque técnico) y la enseñanza de la programación (enfoque lingüístico). Del mismo modo, han convivido diferentes propuestas didácticas dentro de un mismo enfoque. Por ejemplo, la programación se enseñó como una serie de pasos a seguir, que podían ser copiados cual instrucciones (una propuesta más conductista), o como una construcción de acciones para lograr la automatización de una tarea (una propuesta más constructivista). En otros momentos la computadora pasó a ser una herramienta en la escuela, como la calculadora o el manual del grado, o cualquier otro material didáctico que contribuyera a promover los aprendizajes de otras áreas.

Luego de introducida la historia de la enseñanza de la computación en nuestro sistema educativo, se abordaron conceptos lógicos que son centrales a la programación en computación a través de juegos y situaciones problemáticas, sin que sea necesario el uso de una computadora. Los objetivos eran que la resolución de estos problemas y juegos pudieran contribuir a la comprensión de conceptos constituyentes de la disciplina y a desarrollar un modo de pensamiento propio de la computación. A partir del debate de estos conceptos se presentó el concepto de cómputo y máquina universal permitiendo contextualizar el problema que dio origen a la computación y la lógica moderna.

Para finalizar el módulo, lxs docentes tuvieron que organizar una clase de computación a partir de un juego del tipo que realizamos durante el cursado. Además de la planificación, lxs docentes reflexionaron sobre los objetivos que se quisieron lograr con la clase propuesta y las habilidades cognitivas que requirieron de los alumnos teniendo en cuenta el nivel educativo.

Los **objetivos** de este curso fueron:

- Fomentar estrategias de enseñanza por descubrimiento y competencias tales como resolución de problemas, identificación de patrones y abstracción.
- Abordar diferentes problemas lógicos que puedan ser significativos para la enseñanza de conceptos de programación en la Educación Primaria y Secundaria.
- Desarrollar estrategias y heurísticas de resolución y formalización de problemas lógicos y algorítmicos.
- Deconstruir nociones sobre las ciencias de la computación a partir de juegos lógicos.
- Analizar críticamente cómo se ha introducido la computación en la escuela y las implicaciones para el aprendizaje que los alumnos realizan de la disciplina.
- Diferenciar entre TIC y computación.
- Articular conceptos de distintas disciplinas para pensar en proyectos integradores en sus escuelas.
- Diseñar una experiencia de introducción de la programación a través de juegos.
- Debatir en torno al concepto de pensamiento computacional.

Los **contenidos** centrales de módulo fueron:

- La introducción de la computadora en la escuela: paradigmas técnico, utilitario, integrador, y lingüístico. Programas nacionales y provinciales de alfabetización digital.
- Competencias de resolución de problemas lógicos, de ingenio, y de estrategias para desarrollar competencias relativas a búsqueda de variantes, descomposición de un problema en subproblemas, recursión, ordenamiento. Problemas combinatorios. Grafos.
- Estrategias de formalización de las soluciones de los problemas.
- Presentación intuitiva de algunos conceptos de programación para el modelado y la resolución de problemas: variable, estado, acción, secuencia, procedimientos, alternativas. Habilidades de la programación: Abstracciones y representaciones.
- Revisión histórica del nacimiento de las Ciencias de la Computación. El programa de Hilbert. La paradoja de Russell. La máquina universal.

El módulo se desarrolló en dos clases de modalidad virtual y cuatro encuentros presenciales que incluyeron instancias de tutorías. Las clases de modalidad virtual requirieron en su desarrollo dos tipos de tareas:

- instancias optativas orientadas a favorecer la apropiación de los contenidos, no evaluadas.
- actividades de acreditación, cuya realización es obligatoria y serán evaluadas y calificadas.

## Metodología

Para responder a la pregunta de investigación se realizó un estudio mixto que trabajó con la perspectiva cualitativa y cuantitativa. Desde la perspectiva cuantitativa se pudieron analizar datos objetivos tales como retención de la matrícula y valoraciones que lxs docentes ofrecieron del curso en términos de frecuencias.

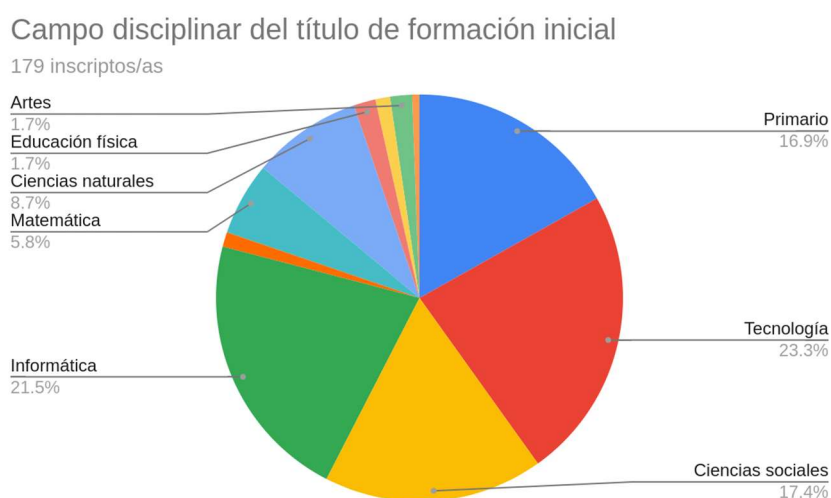
Desde la perspectiva cualitativa analizamos las descripciones de los aprendizajes y los sentidos construidos por los docentes en torno a la disciplina y a los contenidos del curso.

Los datos utilizados provienen de una encuesta de final de cursado con el propósito de aprender sobre la experiencia de lxs docentes con el cursado de este módulo.

## Descripción de la muestra

### ¿Quiénes fueron nuestros docentes?

Para comprender cómo los docentes resignifican los contenidos de este módulo, es necesario primero conocer qué perfiles formativos tiene. Comenzaron la especialización 179 personas, de los cuales 155 finalizaron el primer módulo y 105 egresaron de la especialización. El 65,4% se desempeña en nivel secundario, el 25,7% en nivel primario, el 7,8% en nivel superior y un 1,1% en nivel inicial y especial. La figura 1 muestra los inscriptos y sus formaciones y desempeños previos. En ella se observa que el 21,5% tiene formación en el campo de la informática y el 23,3% tiene formación en el área de las tecnologías. Por otro lado, el 47% (84 cursantes), manifestó tener saberes previos de programación.



**Figura 1:** Campo disciplinar de los y las inscriptas en la especialización

De una encuesta anónima realizada a 175 preinscriptos/as a la carrera elaborada en conjunto con la Fundación Sadosky surge que el 89% considera entre sus las principales motivaciones para cursar la especialización el objetivo de *motivar a sus estudiantes* (89%). En la tabla 2 se muestran estos datos.

Motivación	Cantidad elegida como razón principal	Porcentaje sobre el total de encuestados
Actualización docente	152	86.86%
Carrera docente	89	50.86%
Interés personal	153	87.43%
Para motivar a los estudiantes	156	89.14%
Para darle otros usos al equipamiento disponible	77	44.00%
Actualización de la oferta curricular	114	65.14%
Otro	43	24.57%

**Tabla 2:** Encuesta realizada a 175 preinscriptos/as a la especialización sobre las principales motivaciones para cursar la especialización.

A partir de analizar los datos sobre la cursada al finalizar la especialización, observamos que el Módulo 1 “Lógica, Programación y su enseñanza” fue el más elegido como referencia entre los y

las docentes para realizar su trabajo final de práctica en el aula (28,1%), seguido por los módulos 4 y 6, “Introducción a los lenguajes de programación a través de la animación y los juegos” (25,5%) y “Resolución de problemas de programación a partir del diseño de juegos interactivos” (24,2%), sobre un total de 105 docentes que finalizaron la carrera.

### ¿Cómo fue la cursada del Módulo?

Dado que la enseñanza de las ciencias de la computación es un campo todavía en desarrollo y no se contaba, al momento de comenzar el diseño de la especialización en 2016, con especialistas formados y con experiencia en este campo, se conformó para el diseño y el dictado del módulo un equipo con dos docentes responsables, uno del campo de la informática y otra del campo de la enseñanza, para elaborar los contenidos de este módulo y el dictado de clases grupales, y un equipo de seis (6) docentes (tutores/as), a cargo de seis (6) comisiones, responsables de acompañar individualmente a los y las cursantes y los grupos de trabajo.

De los 179 docentes inscriptos, 155 terminaron y aprobaron la cursada. Como se puede ver en Tabla 3, la tasa de aprobados fue de un 86,6%. El mayor desgranamiento se da al comienzo de la cursada ya que el 7,8% no presenta la primera actividad. Cabe destacar, que en la mayoría de las carreras en el área de informática, se sufre un gran desgranamiento al iniciar la cursada. El desgranamiento aquí presentado, es menor a lo que presentan otras carreras debido principalmente a 2 razones: 1) Se realizó con los preinscritos/as (más de 600 interesados/as) 2 charlas previas obligatorias, para informar sobre los alcances de la carrera y la modalidad de cursada. 2) Se finalizó el proceso de inscripción definitiva el 1er día de cursada, consultando con los cursantes que efectivamente iban a cursar. Esta última acción fue necesaria por razones de cupo, ya que no se contaba con recursos para abrir más comisiones.

		Actividad 1		Actividad 2		Proyecto Integrador (Entrega Final)		Nota Final del Módulo	
Comisión	Matricula Inicial	Aprobados	Tasa de aprobación	Aprobados	Tasa de aprobación	Aprobados	Tasa de aprobación	Aprobados	Tasa de aprobación
Aula A	28	24	85.71%	23	82.14%	23	82.14%	23	82.14%
Aula B	30	29	96.67%	27	90.00%	27	90.00%	27	90.00%
Aula C	32	28	87.50%	26	81.25%	27	84.38%	27	84.38%
Aula D	31	30	96.77%	27	87.10%	26	83.87%	26	83.87%
Aula E	25	24	96.00%	24	96.00%	24	96.00%	24	96.00%
Aula F	33	30	90.91%	30	90.91%	28	84.85%	28	84.85%
<b>TOTAL</b>	<b>179</b>	<b>165</b>	<b>92.18%</b>	<b>157</b>	<b>87.71%</b>	<b>155</b>	<b>86.59%</b>	<b>155</b>	<b>86.59%</b>

**Tabla 3:** Evolución de la cursada

## Resultados

En relación a nuestra pregunta de investigación sobre los aprendizajes de los docentes reconocemos dos campos de saberes. Saberes respecto a la enseñanza de la programación en la escuela y conceptos de programación específicos.

En primer lugar desarrollamos los emergentes que pudimos reconstruir a partir de los datos sobre los saberes respecto a la enseñanza de la programación.

### *Saberes en torno a la enseñanza de la programación*

Un primer saber que se pone en tensión al comenzar la especialización es sobre de qué se trata la computación y en relación a la definición de la computación como campo de conocimiento, cuál es el lugar en el currículum. El paradigma dominante en la escuela es el de la integración de la computación de manera transversal en espacios curriculares propios. Particularmente para el nivel primario.

Sobre una base de 148 docentes que respondieron la encuesta al finalizar el módulo, casi la totalidad sostiene que los mismos pueden ser abordados en la escuela y el 77,7% que se encuentran en parte de las Bases Curriculares actuales y por tanto pueden ser abordados transversalmente o en algún espacio curricular específico (ver Tabla 4).

Este aprendizaje es relevante porque existe la tensión sobre si la computación debe y puede enseñarse para todos los estudiantes o se debería ofrecer (como ocurrió hasta la aprobación de los NAPs de Educación Digital, Programación y Robótica) solamente para estudiantes autoseleccionados ya sea en escuelas técnicas o ciclos de orientación.

Nuestros cursantes, a partir de este módulo aprendieron que los contenidos pueden ser enseñados en el marco de la escuela obligatoria. Como muestra la tabla 4, una gran mayoría sostiene que pueden ser abordados de manera transversal o en espacios curriculares existentes (como matemática o tecnología). Es decir, los y las docentes logran construir una noción de que la escuela puede enseñar programación.

<b>Considerás que los contenidos vistos en este módulo:</b>		
Son abordables desde algún espacio curricular existente como parte de los contenidos previstos en los planes de estudio	48	32.4%
Son abordables de manera transversal como parte de un proyecto curricular institucional enmarcado en el desarrollo de capacidades fundamentales promovido por el Ministerio de Educación	67	45.3%
Son abordables como parte de proyecto curricular institucional enmarcado en el desarrollo de saberes emergentes, no previstos en el diseño curricular provincial	24	16.2%
Son abordables como una propuesta extracurricular	7	4.7%
No son abordables	2	1.4%

**Tabla 4:** Percepción de los cursantes sobre el abordaje de los contenidos del módulo en la escuela.

Asimismo, teniendo en cuenta el énfasis en el paradigma de integración de las TICs que predomina en las escuelas, se observa que los docentes reportan que el módulo logró cambiar su concepción de enseñanza de la computación en las escuelas. La mayoría (52,7%) sostiene que le gustaría incorporar algunos conceptos y/o herramientas didácticas sobre este módulo en sus clases y un 8,8% le gustaría trabajar esos temas en un taller especial. Nótese que este módulo no aborda las TIC, pero precisamente conceptos de lógica y programación. Si bien el 30% reconoce que le faltan experiencias y saberes para aplicarlo en el aula, expresa que el cursado del módulo le permitió pensar en otra dimensión de la computación y las TIC (ver Tabla 5).

<b>El cursado de este módulo, ¿modificó tu concepción acerca de la forma de abordar la enseñanza de la computación en las escuelas?</b>		
Sí, me gustaría incorporar algunos conceptos y/o herramientas didácticas en mis clases	78	52.7%
Sí, me gustaría armar un taller especial para trabajar estos temas.	13	8.8%



Sí, me permitió pensar otra dimensión de la computación y las TIC, pero todavía me falta experiencia y saberes para trabajarlo en el aula	45	30.4%
Sí, pero no me interesa aplicarlo en el aula.	2	1.4%
No, no encontré ningún aporte novedoso.	5	3.4%
Otras	5	3.4%

**Tabla 5:** Percepción de los cursantes sobre el cambio de concepción para el abordaje de la computación en la escuela.

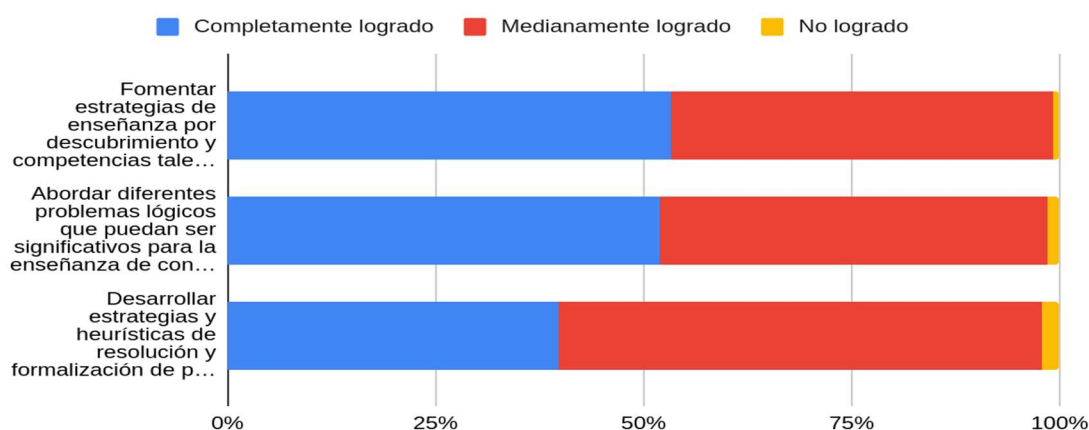
Como se observa en la tabla 5 en relación a nuestra pregunta de investigación, para la mayoría de los docentes este módulo modificó sus concepciones acerca de la enseñanza de la computación. Estos resultados muestran que la perspectiva de abordaje del módulo puede ser de mucha utilidad como un curso introductorio de formación docente. Consideramos que el debate sobre los diferentes paradigmas recorridos en la historia de la enseñanza de la computación en la escuela plantea un punto de partida más fértil para el abordaje de los saberes en la disciplina propia de la computación. A esto se le complementa la introducción a los conceptos de máquina universal y nociones básicas de algoritmos que plantean un nuevo nivel de abstracción para pensar las ciencias de la computación de manera independiente de la máquina que la ejecuta. Además, este recorrido se realiza brindando nuevas herramientas y propuestas para el trabajo en el aula.

### *Saberes en torno a la programación*

Como describimos anteriormente, durante el módulo se ofrecieron ejercicios de lógica para introducir conceptos fundamentales de la programación. Sin bien estas clases pueden ser eficientes en economía de tiempo, no permiten desarrollar las experiencias de aprendizaje significativas de manera que los y las docentes puedan generar experiencias nuevas para sus estudiantes. En este sentido, consideramos que el módulo debe ser complementado con módulos que profundicen en conceptos y en práctica programación que aborden y profundicen nuevos conceptos y permitan desarrollar mayor confianza en la disciplina, incluso en la resolución de problemas lógicos. Podemos ver un reflejo de esta confianza en la evaluación que realizan los cursantes en relación al logro de los objetivos más específicos de la lógica y la computación (Ver Figura 2).

### Valoración del logro de los objetivos del módulo

Encuesta realizada a 149 cursantes al finalizar el módulo



**Figura 2:** Percepción de los cursantes en relación al logro de los objetivos relacionados con la

## lógica y la computación<sup>1</sup>

Se observa en la figura 2 que -si bien la mayoría siente que ha logrado los objetivos- solo la mitad aproximadamente observa que los ha logrado completamente. Asimismo, hay una mayor confianza en saberes relacionados con la enseñanza que en las estrategias propias de la resolución de problemas computacionales.

Cruzando la información con las preguntas abiertas, observamos dos tipos de dificultades o desafíos para el aprendizaje. Uno relacionado a los contenidos de lógica y otro relacionado a la percepción de que los contenidos de historia o filosofía no son transferibles directamente al aula. Por ejemplo, algunos docentes expresaron:

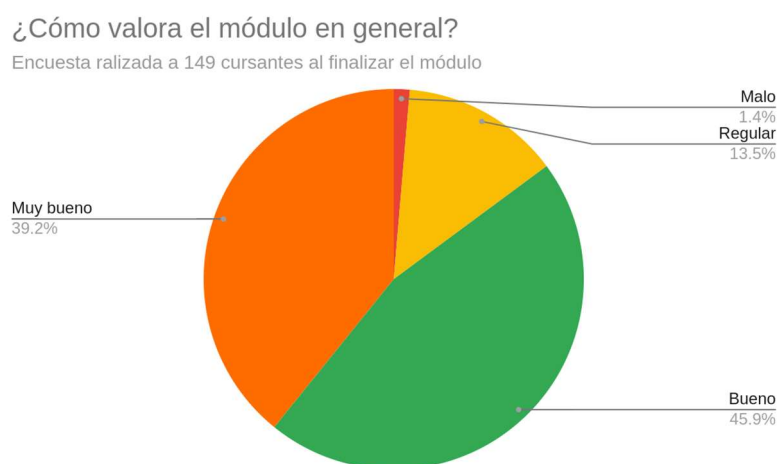
*“Dificultad en la resolución de juegos lógicos en las clases presenciales.”*

*“Demasiada filosofía poco aplicable a la realidad.”*

Estos son dos problemas de aprendizaje diferentes. En relación a los conceptos de lógica el módulo introduce una nueva forma de pensar los problemas en donde es necesario des-semantizar una situación problemática y convertirla en variables que una máquina pueda manipular.

Por otro lado está la cuestión del sentido de los saberes para la enseñanza. Ligado a esto podemos hipotetizar que todavía persiste el paradigma técnico de enseñanza de la computación en donde se enfatiza el conocimiento del funcionamiento de un objeto. En contraste con este paradigma, el módulo ofrecía una versión socio-histórica del contexto de producción de esos objetos que permite dotar de relevancia al hecho técnico. En ese sentido el módulo podría haber explicitado la importancia en la formación de los y las docentes de comprender el contexto de producción para que el aprendizaje de la computación recobre otro sentido.

En relación a la evaluación del módulo en general el 85% respondió que fue bueno o muy bueno.



**Figura 3:** Valoración del módulo por parte de los cursantes

En una pregunta abierta se indagó respecto a la evaluación del curso. La inmensa mayoría de los

1 - Fomentar estrategias de enseñanza por descubrimiento y competencias tales como resolución de problemas, identificación de patrones y abstracción.

- Abordar diferentes problemas lógicos que puedan ser significativos para la enseñanza de conceptos de programación en la Educación Primaria y Secundaria.

- Desarrollar estrategias y heurísticas de resolución y formalización de problemas lógicos y algorítmicos.

docentes respondieron que fue adecuada y permitió recuperar lo trabajado en clase. Asimismo disfrutaron la elaboración de un juego lógico para llevarlo a práctica con sus estudiantes. Algunos docentes expresaron:

*“En este trabajo decidimos aplicarlo a los cursos del ciclo orientado, a mí me tocó trabajar con sexto año y la verdad que capturé su atención logrando una buena participación en la resolución de los desafíos. Los contenidos que desarrollamos fueron datos, algoritmos y patrones, el trabajo en el aula fue grupal. Los resultados fueron muy satisfactorios porque luego entre ellos surgieron los debates para verificar cuales eran los resultados correctos.”*

*“Pudimos observar un interés positivo en los niños que se engancharon con lo propuesto, realizaban intervenciones y lograron los objetivos propuestos.”*

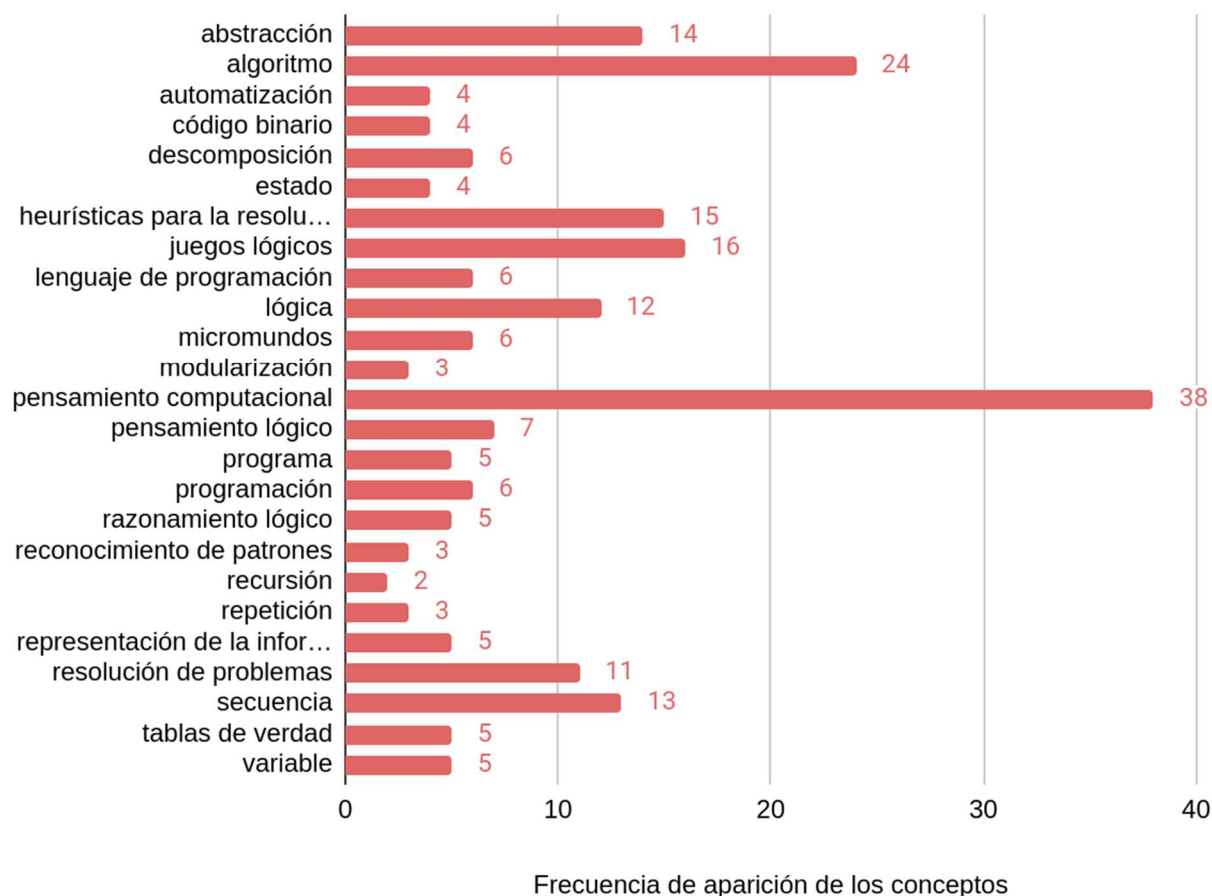
*“Utilice juego lógicos para trabajar el concepto de algoritmo para un primer año (12 años) de las escuelas experimentales PROA, escuelas con énfasis en TIC. Los resultados fueron muy sorprendidos superando expectativas.”*

Es decir, a partir de los datos cualitativos podemos inferir algunas interpretaciones sobre la valoración positiva del curso. Para muchos de los y las docentes el curso tiene valor en tanto pueden observar efectos positivos en los aprendizajes de sus estudiantes. En estas tres citas que tomamos como ejemplo, los y las docentes identifican que los estudiantes participaron, discutieron, atendieron, engancharon y superaron expectativas. Creemos que haber articulado saberes didácticos con los conceptos de lógica pudieron haber contribuido a que los y las docentes diseñaran desde el aprendizaje por descubrimiento, actividades que movilizaran y que produjeran aprendizajes significativos entre el estudiantado.

En relación a los contenidos de programación del módulo, entre las preguntas abiertas, los y las docentes identifican más frecuentemente conceptos de orden general entre los aprendidos tales como: pensamiento computacional, algoritmo y programación. Si bien generales, estos conceptos son relevantes porque son fundamentales a la disciplina, particularmente para romper con los pre conceptos que la mayoría de los y las docentes tenían respecto que saber computación era saber usar programas. Estas ideas aprendidas implican un movimiento en ese sentido donde saber computación se identifica como saber algoritmos, programación y desarrollar un pensamiento particular denominado pensamiento computacional. La figura 4 presenta los resultados.

## Percepción de los cursantes en relación a los conceptos disciplinares aprendidos en el módulo

Pregunta abierta a 148 cursantes que finalizaron el módulo



**Figura 4:** Percepción de los cursantes en relación a los contenidos disciplinares *aprendidos*<sup>2</sup>

Otros conceptos fundamentales de la disciplina que son mencionados con menos son modularización, recursión, secuencia, y variables. Estos con conceptos específicos de programación que se desarrollan con más profundidad en el módulo que continúa al introductorio.

Respecto de los aprendizajes destacamos también que para muchos docentes este módulo permitió cambiar concepciones sobre la enseñanza de la disciplina

*“Como docente de Literatura, esta especialización me abrió la cabeza: me hizo ver que en realidad aplicamos el pensamiento computacional en nuestra vida diaria; muchas veces sin ser conscientes de ello. Incluso en mis prácticas docentes (siii, en Literatura también). Por pensamiento computacional hacemos referencia a*

<sup>2</sup> Al preguntar sobre los contenidos aprendidos, no se incluyen conceptos abordados pero ya conocidos por los cursantes por su formación previa.

*habilidades como: abstracción, representación, descomposición, repetición (iteración), ensayo-error, ordenar datos de manera lógica, etc. Mi profesión me lleva a ser mucho más descriptiva; con lo cual se me dificultan las fórmulas, los algoritmos, seguir cada paso o instrucción sin omitir alguno. Es verdaderamente interesante esto de la programación y el pensamiento computacional. Creo que debe ser incluida como materia obligatoria desde la primaria. Y los juegos lógicos plantean desafíos que despiertan este pensamiento y, lo mejor, atrapan a los chicos. Entonces, por qué no los explotamos más?”*

Esta cita permite identificar cómo los aprendizajes conceptuales de la programación también se relacionan con los aprendizajes sobre la enseñanza de la programación. Conocer el objeto de estudio permite pensar el potencial que tiene para el desarrollo del pensamiento y la necesidad de ser incluida en la escuela.

## Conclusiones

A partir de un análisis de los resultados de las encuestas sobre el módulo “Lógica, Programación y su Enseñanza”, pudimos reconstruir que los y las cursantes desarrollaron al menos dos tipos de saberes: saberes sobre la enseñanza de la programación -su relevancia, su lugar en el currículum, estrategias de abordaje por exploración y descubrimiento- y conceptos fundamentales de orden más general de la programación -algoritmo, pensamiento computacional, variables, secuencias, abstracciones, etc-.

A su vez, consideramos que los objetivos que nos propusimos al incluir el módulo en la carrera fueron satisfechos: 1) presentar el contexto general de la enseñanza de la programación en la escuela, 2) diferenciar las ciencias de la computación de las TIC en general. Además, 3) introducir conceptos preliminares de programación, que son abordados en los módulos siguientes, siguiendo el diseño espiralado de la especialización y 4) introducir herramientas para el abordaje de las ciencias de la computación en el aula.

Consideramos que todavía es necesario la sistematización y el diseño de materiales de formación docente, dada la poca disponibilidad de materiales en esta línea. En este sentido, nuestro abordaje es novedoso y plantea nuevas propuestas para la incorporación en el aula de las ciencias de la computación.

## Referencias

- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). *Developing computational thinking in compulsory education-Implications for policy and practice* (No. JRC104188). Joint Research Centre (Seville site).
- Busaniche, B. (2006). Alfabetización digital: las fronteras del aprendizaje y el control de la Información. *Medios informáticos en la educación a principios del siglo XXI*. Pág, 52, 51-59.
- Denning, P. J., Comer, D. E., Gries, D., Mulder, M. C., Tucker, A., Turner, A. J., & Young, P. R. (1989). Computing as a discipline. *Computer*, 22(2), 63-70.
- Papert, S. (1993). *The children's machine: Rethinking school in the age of the computer*. BasicBooks, 10 East 53rd St., New York, NY 10022-5299.

## Desde los Cimientos

Un estudio sobre la formación inicial del profesorado en relación a la capacitación docente continua en el área de la programación

Silvia Pilar Rodríguez  
ISP "Dr. Joaquín V. González"

### RESUMEN

Desde que la Ley de Educación Nacional del año 2006 especifica entre sus objetivos la inclusión de la tecnología en la escuela, las jurisdicciones se han dedicado a reorganizar sus currículos de manera de cumplir con tal normativa. La resolución 263/15 del Consejo Federal de Educación establece la enseñanza de la Programación como una actividad de importancia estratégica en todas las escuelas del país. En la Ciudad Autónoma de Buenos Aires específicamente, la inclusión de la tecnología en las escuelas se remonta al año 1984, momento a partir del cual pueden encontrarse diversas normativas y acciones tendientes a ese fin (Belnicoff, Consoli, & Lungarete, 2016). La política educativa del GCBA establece que la enseñanza de las TIC en las escuelas sea transversal. Y al respecto modificó también los planes de educación superior incluyendo espacios curriculares que adopten esos contenidos, a la vez que se implementaron instancias de capacitación para los docentes ya en ejercicio, de modo de brindar estos conocimientos para egresados de planes anteriores.

En el presente trabajo se busca comprender la dinámica que se establece entre la formación inicial docente y la capacitación en ejercicio, focalizando en lo que sucede en la Ciudad Autónoma de Buenos Aires y sus institutos superiores del profesorado. Nos preguntamos si tanto una como otra formación, específicamente en lo que se refiere al pensamiento computacional, contribuye a su desarrollo o se trata de un entrenamiento técnico. Y fundamentalmente, si para el desarrollo de esta lógica de razonamiento que implican los lenguajes de programación, un entorno de capacitación podría impactar en la práctica de la misma manera que la formación docente inicial.

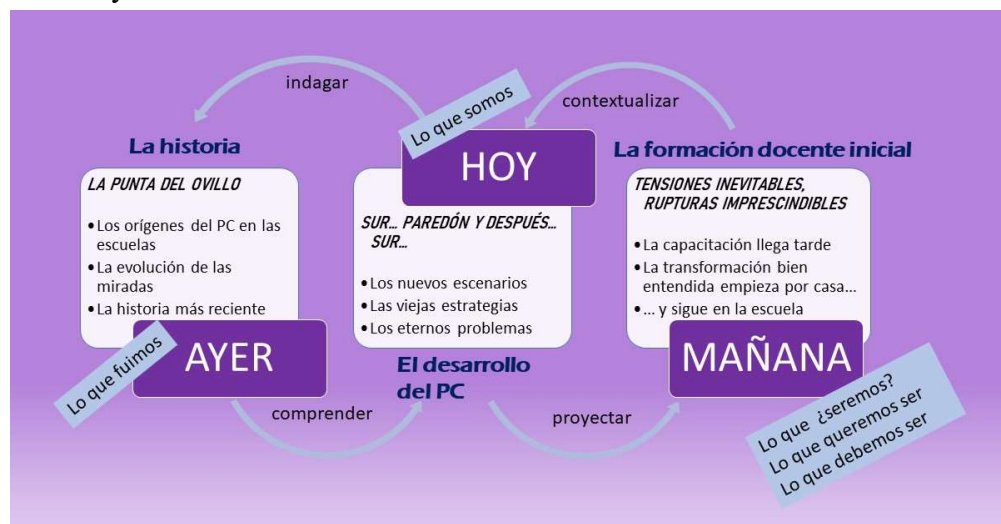
### PALABRAS CLAVE

didáctica de la programación – formación docente – prácticas docentes – pensamiento computacional

### BREVE COMENTARIO SOBRE LA PROPUESTA

La principal preocupación que motiva este trabajo es atender a la necesidad de comprometerse en un análisis muy profundo de la formación docente inicial del presente, en orden a formar futuros profesores que no reproduzcan estrategias didácticas del pasado.

Como podrá observarse, se busca analizar la evolución de la enseñanza de la programación en la escuela desde una perspectiva diacrónica. ¿En qué sentido? Entendiendo que no puede explicarse el hoy sin comprender el ayer, como tampoco puede proyectarse el mañana. Por eso el desarrollo de los ejes de análisis no es lineal. No se propone un recorrido secuencial por los distintos momentos de la enseñanza del pensamiento computacional en la escuela, sino una red entrecruzada de miradas que se revisitan en cada contexto temporal, según el siguiente esquema que muestra esta idea de continuidad y entramado histórico.



## LA PUNTA DEL OVILLO

### *La historia*

Un ovillo se genera a partir de un hilo que se va superponiendo desde su punto inicial. Pero cuando el ovillo ya está armado, encontrar ese origen puede ser una ardua tarea. Se pretende, por tanto, desentrañar esa madeja para abordar la realidad actual de la enseñanza de la programación en la escuela: ¿qué puede aportarnos esta historia del *ayer* para **comprender** la escuela del *hoy*? Y también con la esperanza de ovillar hacia una construcción de estrategias más eficaces para un *mañana* que no tarda en llegar.

## SUR... PAREDÓN Y DESPUÉS... SUR...

### *El desarrollo del PC*

La maravillosa poesía de este conocido tango nos invita a detectar rupturas que en el fondo no hacen más que generar continuidades: más allá de Pompeya, al sur, detrás del paredón cruzando el Riachuelo, solo hay más sur.

Tomando esta metáfora, se analiza la situación actual de la enseñanza de la programación y el desarrollo del pensamiento computacional en las escuelas. ¿Cuál es el verdadero aporte que esta enseñanza conlleva? ¿Hay alguna modificación de las prácticas en virtud de estos aportes? ¿Existe algo distinto después del paredón, al sur del sur? ¿O se trata de las mismas prácticas de siempre con distintas herramientas?

Se busca recorrer este tramo temporal permitiéndonos pararnos en el *hoy* para **indagar** el *ayer* y **proyectar** un *mañana* de superación.

## TENSIONES INEVITABLES, RUPTURAS IMPRESCINDIBLES

### *La formación docente inicial*

Este tramo es el nudo del trabajo. Se pretende aquí descubrir las tensiones que se manifiestan en la formación docente inicial, considerando que la misma es base indiscutible de la escuela que queremos. Entonces, ¿Cómo estamos formando a nuestros futuros docentes? ¿La formación del profesorado en el área de la programación tiene en cuenta la escuela del presente y sus objetivos? ¿Y la del futuro? ¿Hay una mirada superadora de las viejas didácticas?



Desde esta postura, se distingue la “formación inicial” de la “capacitación docente continua”, la cual es imprescindible para la constante actualización profesional, pero las bases sólidas de una formación académica solo se consiguen en los institutos donde se desempeñan los formadores de formadores, como agentes multiplicadores por excelencia. Luego sí, la formación docente continua en el ejercicio profesional, debe proveer una actualización permanente. Se trata entonces de **contextualizar** esta formación docente en el *hoy* de la escuela, como producto de un *ayer* que ha evolucionado, y que permite construir la escuela de *mañana*.

## INTRODUCCIÓN

En virtud de la normativa mencionada anteriormente, surgieron en los últimos años muchas iniciativas que apuntan a la enseñanza de la robótica y el pensamiento computacional en las escuelas en sus tres niveles: inicial, primario y secundario. Instituciones como la Fundación Sadosky<sup>1</sup>, por ejemplo, han llevado adelante numerosas tareas en este sentido: capacitaciones docentes, prácticas situadas, investigaciones, desarrollos de lenguajes y entornos de programación, etc. Cabe preguntarse entonces, qué relación hay entre estos planes y la formación inicial docente en el área de programación. Mientras las capacitaciones intentan modificar las prácticas de los docentes en ejercicio, la formación inicial debe aportar al sistema los profesionales que se requieren para desempeñar esa función. De esta manera los egresados de los profesorados, futuros docentes de nivel medio, deberán hacerse cargo de un área que apunta a un objetivo propedéutico: el desarrollo del pensamiento computacional. El esfuerzo que las instituciones escolares realizan atacando los síntomas de una formación deficiente y obsoleta (o al menos insuficiente y pensada para otro contexto), debería ser acompañado por la participación activa de los institutos superiores del profesorado, asegurando así que las causas de tal formación no se sostengan ni se multipliquen.

Las capacitaciones docentes en pensamiento computacional en el ámbito de la CABA son, en algunos casos, técnicas que los profesores y las profesoras de diversas áreas aplican para llegar a un resultado. Desde este punto de vista, el “pensamiento” -en cuanto a reflexión crítica sobre los sucesos involucrados en un programa- parece perder protagonismo para dar paso al “éxito” en el funcionamiento del programa. Estos cursos refuerzan la idea de alienación tecnológica por la cual los sujetos utilizan los sistemas sin comprender lo que subyace a los mismos, con lo que permanecen en el desconocimiento de lo profundo (Quintanilla, 1989). Más aún, en algunos casos la metodología con la que se imparten estas capacitaciones conserva el marco de la didáctica tradicional: enseñanza teórica – práctica – evaluación. Existen también experiencias de formación continua que se enmarcan en otros dispositivos didácticos, pero cuando el docente ya se encuentra en el ejercicio de su profesión, tras largos períodos llevando adelante un abordaje áulico tradicional, se puede incluso llegar al rechazo de estas prácticas, haciendo que el curso de capacitación tenga el efecto contrario al deseado.

Si se pudiera apuntar a la formación inicial docente anticipándonos a esta problemática, generando, análogamente al concepto de vacuna, una estructura docente inmune que le permita al futuro profesional tener otra mirada acerca de las tecnologías, probablemente la formación continua pudiera tener otra relevancia. Para ello, en los institutos del profesorado, se debería favorecer la apropiación de las tecnologías, lo que Quintanilla denomina tecnologías entrañables (Parselis, Mayo 2016). Este es un problema central para el profesorado de Informática, dada su especificidad, pero se trata de un problema metodológico para la docencia en general.

## VINO NUEVO EN ODRES VIEJOS

Mucho se habla de la formación docente continua y la importancia de profesionalizar la docencia. No cabe duda de que debe ser así, que el docente es un profesional en constante aprendizaje y que el conocimiento no puede tener una meta de llegada sino que constituye siempre un punto de partida. No obstante cabe preguntarse: ¿Cuán efectiva puede ser la capacitación docente en ejercicio

<sup>1</sup> Información sobre este particular puede encontrarse en <http://www.fundacionsadosky.org.ar/formacion-de-capacitadores-para-la-programacion-y-su-didactica/>



si difiere mucho de la formación inicial? ¿Se puede realmente instalar nuevas prácticas educativas en las escuelas si estas se alejan mucho de las que se imparten en los profesorados?

En muchas escuelas de la CABA y en los distintos niveles se están llevando a cabo cursos de capacitación a docentes de todas las áreas en lo relativo al pensamiento computacional. Se pretende que a través de ellos, los profesores de biología, por ejemplo, desarrollen con sus alumnos un sistema automatizado de riego, que los profesores de educación cívica realicen simulaciones de elecciones democráticas, o que los profesores de música inviten a sus alumnos a componer digitalmente. Esto se da bajo el lema: la tecnología debe ser un contenido transversal. Más allá de los (des)acuerdos que podamos establecer con esta premisa, ¿es esto posible?; ¿están dadas las condiciones en los establecimientos educativos para responder a esta demanda?

Coincidimos con Andrea Alliaud (Alliaud, 2004) en que el maestro que recién se inicia no es una tábula rasa, un profesional sin ninguna experiencia; lleva consigo al menos trece años de rutina escolar. Si a estos le agregamos los cuatro o cinco años que demanda una carrera docente, se constituyen en casi veinte años frente a las mismas prácticas (Alliaud, 2007). Consideramos por tanto imprescindible que la formación docente inicial persiga como objetivo reflexionar sobre ellas y analizarlas críticamente. ¿Sucede esto en los profesorados de la CABA? ¿Se abordan todos los espacios de formación con esta premisa? ¿Alcanza con un espacio dedicado a la tecnología para que se incorpore verdaderamente esta forma de trabajo para el logro de los objetivos?

Pongamos como ejemplo el modo en que los estudiantes aprenden a resolver triángulos mediante funciones trigonométricas. Quienes pasamos por el nivel medio podremos acordarnos, más o menos clara o vagamente, los esquemas en los que se nos daba como dato un ángulo de un triángulo y las medidas de algunos de sus lados; un dibujo que seguramente encontraremos hurgando un poco en la memoria. Había que analizar si ese problema tenía datos suficientes para aplicar teorema del seno o del coseno, y en base a eso, recurrir a la fórmula, reemplazar los datos y resolver matemáticamente. Todos y todas en algún momento lo resolvimos, de lo contrario no tendríamos nuestras acreditaciones. Sin embargo: ¿cuántos de nosotros recuerda ese tema? ¿cuántos serían capaces de explicar cuál es la utilidad práctica de semejante tarea? ¿Por qué es importante que un estudiante de nivel medio sepa resolver estos problemas para poder exhibir su diploma? Estas preguntas podrían ser más difíciles de responder que el problema en sí mismo...

Desde el punto de vista de la didáctica de la matemática surge una nueva pregunta: ¿por qué en las escuelas se enseña este tema de esta manera? Podríamos asegurar, sin temor a equivocarnos, que contestar semejante cuestión solo sería posible luego de una investigación seria al respecto. Sin embargo, y como hipótesis anticipatoria basada en trabajos previos sobre el tema (Palacios, 2011), nos animamos a aventurar que hay algo de la biografía escolar del profesor de matemática que de algún modo condiciona sus prácticas. Quien durante toda su vida escolar aprendió de determinada manera, tenderá a repetir estas prácticas en el ejercicio de la docencia. Así lo expresan numerosos trabajos de investigación.

Al respecto, Gloria Edelstein realiza un análisis de las prácticas de enseñanza. En el capítulo 5 de su libro, la autora señala la diferencia entre las macrodecisiones y las microdecisiones a la hora de resolver cuestiones áulicas: las primeras se enmarcan en una teoría didáctica y pedagógica que en general es abordada en los profesorados, mientras que las segundas refieren a situaciones muy particulares y del cotidiano escolar, que están más condicionadas por la propia experiencia, y no encuentran solución en las “anticipaciones hipotéticas” de las macrodecisiones. Es aquí donde la experiencia juega un papel fundamental, y esa experiencia apela a la trayectoria autobiográfica del docente, encontrándose múltiples particularidades en relación a la disciplina de base (Edelstein, 2011).

Ahora bien, si lo que vamos a analizar es la formación docente inicial, y volviendo a nuestro ejemplo, la pregunta que cabe es: ¿cómo aprenden estos temas los estudiantes del profesorado de matemática? La respuesta es: en principio lo ignoramos, pero lo que sí podríamos afirmar es que si en las casas de estudios superiores la modalidad de enseñanza es la misma que la que vienen teniendo

los niños y adolescentes en las escuelas, muy probablemente ese futuro docente se incline por enseñar a resolver triángulos de la misma manera que lo hicieron nuestros profesores hace veinte, treinta o cuarenta años.

Se podrá argumentar que en los profesorados no solo se enseñan las materias disciplinares; hay toda una batería de materias que ponen el acento en lo didáctico y lo pedagógico y que les dan a los estudiantes la posibilidad de contar con una amplia gama de herramientas a la hora de desempeñar su tarea. Y eso es real, pero la pregunta es: ¿es suficiente para que luego pueda trasponer ese saber a la enseñanza de un tema que aprendió de una sola manera a lo largo de toda su vida estudiantil?

Se toma el ejemplo de los triángulos porque es un ejercicio que seguro todos hicimos, independientemente del momento en que cursamos nuestra escuela media. Pero esto naturalmente es válido también para la enseñanza y el aprendizaje de la programación, una disciplina que en los últimos años se masificó en las escuelas argentinas: si la enseñanza es la transmisión de una serie de pasos para resolver un problema sin el debido razonamiento sobre los mismos, así aprenderán los estudiantes; y si esta modalidad se mantiene en el profesorado, así enseñarán los docentes. Y con una base tan fuerte ¿es esperable que los cursos de capacitación logren transformar las prácticas? Es esta la pregunta que guía nuestra reflexión...

### UNA HISTORIA PENDULAR

Es curioso observar el derrotero que fue realizando la enseñanza de la computación en las escuelas de la CABA. Como un péndulo que va de un extremo al otro hasta que finalmente encuentra su equilibrio, el pensamiento computacional fue ensalzado y denostado hasta su ubicación como materia transversal a todo el curriculum. Resta saber si este es un punto equilibrado o no...

En la década del '90 algunas escuelas privadas de la ciudad, fundamentalmente de elite, incorporaron a su oferta extracurricular la enseñanza del Logo, un lenguaje de programación pensado para los niños que, a modo de juego, debían encontrar las instrucciones precisas para realizar un dibujo en la pantalla. El objetivo de tal actividad era fomentar la curiosidad, desarrollar el pensamiento lógico y estimular la creatividad (Feurzeig, 1984). El entorno de trabajo era muy poco amigable, recordemos que para esa época las PC no eran un elemento habitual en los hogares y su sistema operativo DOS requería conocimientos específicos de operación de comandos. Se trataba de un dibujo que simulaba una pequeña tortuga (por eso se lo conoce como “el programa de la tortuguita”) a la que había que decirle con mucha precisión cómo moverse por la pantalla para que dibujara lo que se deseaba. El juego fue un furor entre los niños, pero muy pronto resultó repetitivo y terminó por aburrir a docentes y estudiantes.

Más adelante, pero aún bajo el mismo sistema operativo, se difundió en muchas escuelas el uso de un video juego que favorecía el desarrollo del pensamiento basado en estructuras de programación (decisiones simples y múltiples, sentencias de control, ciclos, manejo de variables) llamado Maniac Mansion, en el cual el objetivo era, a través de órdenes precisas a modo de algoritmo y toma de decisiones, rescatar a una joven encerrada en una mansión por un malvado villano. Para ello, cuatro compañeros debían ir resolviendo pistas y encontrando objetos que posibilitaran el acceso a la torre de encierro. Los dibujos y sus movimientos eran muy primitivos, no tenía sonido sino que los distintos personajes expresaban los mensajes mediante globos de diálogo, y las instrucciones debían seleccionarse por teclado en una serie de opciones en la parte inferior de la pantalla. A pesar de esta precariedad, muchos niños y niñas se entusiasmaban y se esforzaban por “rescatar a la joven”, y esto los movilizaba a poner en ejercicio una serie de estrategias de acción que fomentaban el razonamiento lógico: las sentencias se formaban mediante una estructura sintáctica “verbo (acción) + objeto directo (variable)”; como en los lenguajes de programación de alto nivel, seleccionar la acción adecuada para la variable correcta era lo que permitía ir diseñando el algoritmo para llegar al objetivo (abrir las distintas puertas).

Mucho más conocido que el anterior es el famoso “Carmen San Diego”, un video juego perteneciente a esa misma época que consistía en atrapar a un astuto ladrón que escapaba por todas

partes del mundo, con lo cual no solo había que poner en juego estrategias de pensamiento computacional para la correcta toma de decisiones, sino también tener conocimientos de historia y geografía. Fue muy utilizado en las escuelas durante la mencionada década.

Paralelamente a la situación escolar, la utilización de computadoras avanzaba también en los lugares de trabajo y, en menor medida, en los hogares. En las casas donde se contaba con una computadora, los padres debían establecer horarios para organizar el uso de ese único elemento por parte de todos los miembros de la familia, y los niños esperaban el momento de acceder a él para consultar una enciclopedia, escribir algún trabajo práctico... pero también para jugar. Sí, la computadora parecía ser un gran distractor de las obligaciones escolares de los pequeños. Los video juegos fueron, en consecuencia, demonizados y asumidos como enemigos de la educación, en tanto atraían la atención de los niños y adolescentes que dejaban a un lado el estudio. Esa misma lógica familiar fue trasladada a las escuelas cuyas autoridades, muchas veces desconocedoras de las virtudes de algunos programas con los que los niños aprendían al tiempo que jugaban, terminaron prohibiendo su uso.

Esta misma década del '90, como dijimos, el uso de la computadora se volvió bastante habitual en bancos, empresas y hasta comercios. Surgieron así la lectora de código de barras en los supermercados, los cajeros automáticos en las sucursales bancarias, las tarjetas de crédito con bandas magnéticas que debían ser leídas en los comercios y transmitidas vía telefónica a las centrales. Y empujado por el advenimiento de este hardware, surgieron también nuevas necesidades de software, en especial, la gestión de bases de datos, que se agregaba a los ya conocidos utilitarios de procesamiento de texto y de planillas de cálculo, conformando el "Paquete de oficina", más conocido por su nombre comercial. Esto último no es casual, y también condicionó el curriculum. Pese a que existían muchas empresas dedicadas a la producción de software de oficina, Microsoft impuso su marca tan fuertemente que no se tenían en cuenta otras opciones, ni siquiera aquellas que, respondiendo a la filosofía GNU de desarrollo de software (Stallman, 2004), volcaban sus productos al mercado en forma libre y gratuita. Es por eso que en las escuelas lo que se enseñaba era, en definitiva, el "Paquete Office".

En este nuevo contexto de evolución tecnológica en el mercado laboral, las escuelas se transformaron en escenario de entrenamiento de los "trabajadores del siglo XXI" que, desde una mirada utilitarista de la educación, se dedicó a formar egresados de nivel medio que tuvieran conocimientos de software de oficina y manejos contables. Es así como muy pronto el curriculum escolar referido a la enseñanza de la computación, se limitó a un procesador de textos, una planilla electrónica, un software de gestión de bases de datos y un software de presentaciones. En los primeros grados estos se integraban (o al menos se intentaba) a los temas curriculares del curso; así, por ejemplo, se hacían presentaciones sobre San Martín en el mes de agosto, cuentos en el procesador de textos aplicando lo estudiado en lengua, cálculos matemáticos y estadísticos en las planillas electrónicas, etc. Particularmente, la escuela secundaria de la CABA reemplazó la materia "Actividades Prácticas" por "Computación", que se dictaba en forma obligatoria en primero y segundo año, y en forma optativa (los estudiantes podían elegir entre Computación y Educación Plástica) en tercer año.

Luego de la sanción de la Ley de Educación Nacional, en el año 2006, la mirada sobre estos temas volvió a poner el foco en el pensamiento computacional, y la enseñanza de la robótica fue la estrella de los últimos diez o doce años. La variante en este caso, fue que este modo de razonamiento debía darse transversalmente en todo el curriculum. Y sobre todo, que trascendiera las fronteras de la escuela de gestión privada, y se universalizara a todas las escuelas públicas de la ciudad. Así nacieron proyectos de huertas con riego automatizado, construcción de dispositivos de accesibilidad, simulación de procesos químicos, y otras actividades que, utilizando la robótica como herramienta, daba solución a problemas de las más diversas áreas de estudio.

Esta visión de la computación como disciplina transversal tiene la ventaja manifiesta de amalgamarla con todas las demás materias: la tecnología, y más precisamente la programación,

pueden verse con claridad como herramientas al servicio, en definitiva, de la vida humana. Pero como contrapartida, tiene la dificultad de no contar con un espacio liderado por especialistas; resulta difícil aventurar cuánto podría un docente de Historia, saber sobre programación como para poder conducir un proyecto en ese sentido. Este debate que se dio a principios de la década pasada, tuvo su impacto sobre las decisiones de política educativa, y redundó en la inclusión de materias disciplinares relativas al tema en tercero, cuarto y quinto año de las escuelas de nivel medio de la CABA (Cucuzza, 2015).

A pesar de tal inclusión, el objetivo seguía siendo implementar proyectos interdisciplinarios en todos los cursos, incluso en el nivel primario. Para ello se diseñaron numerosos espacios de capacitación que apuntaban a preparar a los docentes de cada área en la enseñanza de la programación. La Fundación Sadosky fue pionera en el dictado de estos cursos, liderando en la Argentina, un interesante debate que se daba a nivel mundial.

### ENTRE USAR Y PENSAR

Todos los sistemas educativos iniciaron un camino de reflexión y debate sobre la enseñanza de la programación en las escuelas, y el lugar que debía ocupar el desarrollo del pensamiento computacional en esta enseñanza. No se trataba solo de utilizar la tecnología en la escuela, sino fundamentalmente de pensarla.

En España, por ejemplo, el conocido especialista en tecnología educativa Jordi Adell, lo expresa en términos políticos, ya que existen dos perspectivas que defienden el desarrollo del pensamiento computacional en la escuela obligatoria: la que aborda los contenidos con el objetivo de adquirir habilidades para desempeñarse en el mercado laboral, y la que postula el pensamiento computacional como una manera de desarrollar las capacidades cognitivas de los estudiantes. Pero ninguna de ellas contempla *“la necesidad de dotar a los ciudadanos y ciudadanas de los conceptos, habilidades y actitudes precisas para desarrollar una competencia digital crítica que les permita afrontar su proyecto de vida en una sociedad”*. Y agrega que *“comprender qué es un algoritmo requiere también comprender por qué nuestra vida y nuestras relaciones, por ejemplo, con la administración y el poder, están cada vez más mediatizadas por sistemas algorítmicos, en qué tipo de supuestos se basan dichos sistemas, qué consecuencias tienen en nuestras vidas y qué derechos debemos reivindicar”* (Adell Segura, Llopis Nebot, Esteve Mon, & Valdeolivas Novella, 2019, pág. 181).

Aquí en nuestro país, la Universidad Nacional del Comahue en convenio con el Ministerio de Educación de la provincia de Neuquén, han realizado algunas investigaciones tendientes a establecer cuál es el lugar que ocupan las ciencias de la computación en el curriculum escolar (Branchini, Cortez, Pedemonte, & Rodríguez, 2019), y más precisamente, en la escuela media (Sommer, Cornejo, Cortez, & Rodríguez, 2018). En estos trabajos se pretende identificar enfoques, paradigmas, perspectivas y alcances de la enseñanza de las ciencias de la computación en las escuelas argentinas, y qué relación tiene esta enseñanza con el desarrollo de competencias digitales y de pensamiento computacional en los estudiantes.

Dentro de estos marcos de debate planteados, se ve claramente que el objetivo va más allá de una necesidad utilitarista de resolver un problema determinado mediante una solución tecnológica, sino que apunta a profundizar en el desarrollo de los procesos intelectuales que son necesarios para alcanzar esa solución. Teniendo en cuenta esto, cabe preguntarse si los proyectos que se realizan en las escuelas involucran el pensamiento computacional o simplemente “copian” porciones de código que hagan funcionar los robots, cuestión que estudiaremos en el siguiente apartado. Si así fuera, demás está decirlo, no habría desarrollo de razonamiento lógico alguno.

### EL CAMINO COMO PUNTO DE LLEGADA

Párrafos más arriba hacíamos mención a la forma en que la trayectoria escolar del docente condiciona sus prácticas, ejemplificando con un tema relativo a la enseñanza de la matemática por tratarse de un área que todos quienes hicimos la escuela media, independientemente de nuestra edad,

transitamos. Retomemos ahora esa idea aplicada a la enseñanza de la programación como disciplina que potencia el desarrollo del pensamiento computacional.

Sabemos que la tecnología en general, y la producción de software en particular, han sufrido en sí mismas cambios muy profundos en muy poco tiempo. Hace apenas cincuenta años aún se programaba utilizando tarjetas perforadas en código binario, una tarea reservada a unos pocos especialistas que se preparaban años para desempeñarla. Actualmente, las interfaces visuales, sumadas a la disponibilidad de tutoriales en la web, permiten que cualquier persona con un poco de curiosidad diseñe una aplicación. Entonces cabe preguntarse: ¿por qué es importante, y las organizaciones así lo recomiendan, enseñar programación en las escuelas?

La escuela de nivel medio no es, excepto en la modalidad técnica, una institución que apunte a la formación de profesionales. Por lo tanto el objetivo no debería ser formar futuros programadores para desempeñarse en el área del desarrollo de software. Por un lado, es interesante que los estudiantes conozcan las potencialidades de la programación en orden a ofrecer un abanico de posibilidades de dedicación a futuro. Por el otro, y aquí está el punto que lo hace verdaderamente interesante y especial, pensar desde cero cómo diseñar un algoritmo pone en juego ciertas habilidades de razonamiento que, ejercitándolas, apuntan a desarrollar en el estudiante el pensamiento computacional.

Teniendo en cuenta esto último, lo que se desea perseguir es que el niño (o el adolescente, hablando de escuela media) haga todos los procesos de razonamiento que lo lleven a resolver el problema, evaluando alternativas de solución, definiendo variables, decidiendo estructuras para aplicar, en fin, pensando con lógica computacional. Y esto se traslada no solo a la disciplina en sí misma, sino a todas las áreas que involucran el conocimiento y que en mayor o menor medida, se apoyan en la tecnología para resolver sus problemas. Es decir, no se busca el punto de llegada (resolver el problema en cuestión), sino más bien poner el acento en el camino (la lógica de razonamiento).

Puesto de manifiesto el objetivo, los contenidos y las estrategias didácticas deben ser tales que dicho objetivo pueda alcanzarse. Si nuestro objetivo fuera, por ejemplo, automatizar el riego de una huerta, armaríamos la estructura de hardware con los sensores y actuadores correspondientes, y escribiríamos las instrucciones necesarias para que tal sistema funcione. No importa si esto lo hacemos pensándolo desde cero o copiando códigos que ya están armados para cada dispositivo y compartidos en la web: el objetivo es que el sistema de riego automático funcione. Pero si lo que queremos es propiciar el desarrollo del pensamiento computacional en los niños y adolescentes, lo importante son todos los procesos que están involucrados en la producción de ese software, más allá de su correcto funcionamiento. Entonces, las prácticas docentes deberían dirigirse a este objetivo.

En este esquema de organización del curriculum, se apunta, como se dijo, a que profesores de todas las áreas se involucren en la enseñanza de la programación. Siguiendo el ejemplo anterior, es el profesor de biología quien debería llevar adelante el proyecto. Sin embargo, este profesional, salvo casos puntuales en que así lo haya decidido, no cursó estudios que le permitan abordar estas estrategias y estos contenidos, y en consecuencia, sus prácticas podrían tender a solucionar el problema a como dé lugar, con lo cual lo que habitualmente termina sucediendo es la transcripción literal del código que se vio en los cursos de capacitación docente, sin una verdadera elaboración del pensamiento. En cambio, sería de esperar que los profesores que se encuentran actualmente cursando su formación docente, sí tuvieran este bagaje de conocimiento que les permitiera un abordaje más adecuado del tema y una práctica más acorde con los objetivos que se persiguen. Que esto sea así depende de los institutos de formación docente.

Por eso volvemos sobre las preguntas que son el eje de reflexión de este trabajo: ¿es suficiente con la capacitación docente para alcanzar este objetivo? ¿qué lugar ocupa la programación en la formación inicial del profesorado?

## UN BUEN PUNTO DE PARTIDA ASEGURA EL CAMINO

La capacitación docente continua apunta a la actualización tanto de contenidos disciplinares como de estrategias didácticas que van surgiendo a lo largo de múltiples investigaciones, y que en la formación inicial de esos docentes no se conocían. Tal es el caso de la capacitación en tecnología: una rama del conocimiento que, como explicamos en el apartado anterior, hace muy poco se hizo masiva y se enseña en todas las escuelas. Es de esperar que el grueso de los docentes no esté preparado para trabajar con sus alumnos en este sentido. El Ministerio de Educación de la Ciudad Autónoma de Buenos Aires considera como una prioridad a las capacitaciones en tecnología para los docentes en ejercicio. Así lo expresan las autoridades de tal organismo a través de la página web institucional: *“teniendo como misión reivindicar el rol profesional del docente, apostando por la innovación en investigación y prácticas de enseñanza y aprendizaje, en base a las necesidades y los desafíos del Siglo XXI. Tiene como objetivo brindar capacitación docente de excelencia, gratuita y directamente vinculada a las temáticas y problemáticas educativas vigentes, para favorecer la mejora integral del sistema de educación en la Ciudad Autónoma de Buenos Aires”*<sup>2</sup>

En otras disciplinas, la formación docente continua apunta a nuevas miradas de los contenidos, nuevas estrategias didácticas y, por supuesto, el modo en que la tecnología ayuda al aprendizaje de los temas relativos a esa materia. Es así como podemos observar que la transversalidad de la enseñanza de tecnología está presente también en ese recorrido profesional. Entonces nos preguntamos... ¿es así también en la formación inicial?

Veamos...

Los planes de las distintas carreras de profesorado en la CABA han sido modificados en los últimos tiempos. Y esta modificación se debe, en parte, a la necesidad de incluir en el curriculum, contenidos que tienen que ver con las tecnologías. No solo en lo estrictamente técnico, sino también en las implicancias de estas tecnologías en la sociedad y en las subjetividades. Así es como ahora encontramos espacios curriculares tales como “Nuevas tecnologías” y “Nuevos escenarios”, apuntando el primero a una visión instrumental y el segundo con una perspectiva más reflexiva y de puesta en debate de la situación actual.

Pero ¿es esto suficiente? ¿apunta a los mismos objetivos que los cursos que se dictan en las instancias de formación docente continua? ¿qué vinculación tienen esos espacios con la didáctica específica de la disciplina del profesorado en cuestión? ¿alcanza con mencionar la tecnología y reflexionar sobre ella uno o dos cuatrimestres? ¿propicia esta forma de enseñanza un cambio en las prácticas docentes? Todas estas preguntas merecen un trabajo de investigación serio que las conteste. A priori, podríamos decir que son pertinentes, que no es descabellado hacerlas, aunque no podemos conocer las respuestas a ellas. Sin embargo, nos permitimos una reflexión al respecto.

Si lo que se busca es instalar otra forma de hacer docencia, estimular la utilización de otras didácticas, plantear nuevas prácticas, resulta imprescindible contar con una formación docente continua que pueda ayudar a llevar adelante estos desafíos. Una formación docente continua que nos acompañe en un camino nuevo, poco explorado y desconocido para muchos que hemos tenido otra formación en nuestros profesorados, y que no está acorde con los tiempos de estos nuevos escenarios.

Ahora bien, si la formación docente inicial no revisa también sus propias prácticas docentes, ¿no es de esperar que la formación continua sea mucho más difícil? O dicho de otro modo ¿no sería mucho más efectivo que en los profesorados se aplicaran estas mismas prácticas docentes por parte de los formadores de formadores? ¿no sería más fácil después el camino de la formación docente continua y mucho más aceptadas sus propuestas?

Volvemos a nuestro planteo inicial: si el profesor de matemática aprendió a resolver triángulos siempre de la misma manera, tanto en su escuela secundaria como en el profesorado, será muy poco probable que luego en el ejercicio de su profesión aplique otra metodología de enseñanza, aun cuando

---

<sup>2</sup> Este texto fue extraído del sitio web oficial del Ministerio de Educación del Gobierno de la Ciudad de Buenos Aires. Disponible en: <https://www.buenosaires.gob.ar/educacion/docentes/escuelademaestros/quienes-somos>



las autoridades del ministerio inviertan recursos en llevar adelante cursos de actualización que promuevan un cambio de estrategia didáctica.

### LA TEORÍA DEL DERRAME... DE LAS PRÁCTICAS DOCENTES

En 1776, mientras el Virreinato del Río de la Plata veía sus primeras luces, un escocés llamado Adam Smith escribía “La riqueza de las naciones”, una obra que le valió el título de “padre de la economía política”, disciplina que hasta ese momento no existía como tal. En esa obra detalla que tal riqueza debe construirse en base a tres pilares básicos: el crecimiento económico, la división del trabajo y la libre competencia. Según él, la “mano invisible” del mercado era la encargada de acomodar las variables para el desarrollo, y cada inversor, cuidando sus propios intereses, contribuiría a favorecer el bienestar de todos: “*no es la benevolencia del carnicero o del panadero la que los lleva a procurarnos nuestra comida, sino el cuidado que prestan a sus intereses*”. Esta teoría fue reformulada, dos siglos más tarde, por Milton Friedman, un economista monetarista que en su libro “Capitalismo y libertad” retomó los conceptos de Smith y expresó que ese crecimiento que él postulaba redundaba en beneficio de la sociedad en su conjunto a través del “tricking down effect” o “efecto cascada”, lo que en Argentina se popularizó con el nombre “Teoría del derrame” aludiendo a la “*metáfora de un vaso que primero debía llenarse, para luego poder derramar el ‘excedente’ al exterior (es decir, hacia los más desfavorecidos)*” (Fair, 2010).

Huelga decir aquí que esta teoría cuenta con numerosos detractores que han dado pruebas de la falsedad de la misma, y la propia historia de nuestro país se ha encargado de derrumbarla. Sin embargo, la metáfora del vaso que debe llenarse para luego derramar puede ser útil a la hora de describir lo que se propone este trabajo. Los invitamos, por tanto, a despojar a tal expresión de las fantasmáticas que la rodean...

¿En qué sentido se propone aquí una “teoría del derrame”? Dicho así parece hasta contradictorio, más aún hablando de tecnologías educativas que promueven la construcción colectiva y colaborativa de conocimiento. Pero con este exabrupto, si se nos permite, queremos mostrar que todo cambio de prácticas docentes será mucho más efectivo si comienza por las prácticas en las clases del profesorado. Si los profesores a cargo de los espacios curriculares disciplinares de las carreras docentes pusieran en juego una amplia variedad de estrategias didácticas renovadas, sería mucho más factible la implementación de estas en las aulas de las escuelas secundarias, una vez que los estudiantes obtengan su título y desarrollen su tarea docente. No estamos queriendo decir, en lo absoluto, que el conocimiento pertenezca a un conjunto de iluminados y que desde ahí simplemente se “derrama” sobre los estudiantes, sino más bien que las prácticas docentes se hacen más efectivas cuando se robustecen desde la formación inicial del profesorado, en lugar de buscar corregirlas luego mediante la formación docente continua. Parafraseando la teoría de Fair, deberíamos asegurarnos que ese vaso sea lo suficientemente amplio, flexible y adaptable como para contener variedad de fluidos que luego puedan ser administrados en las clases. Si el vaso está preparado solo para contener el líquido que se administra inicialmente, es probable que no acepte otro, o lo acepte parcialmente, o se deberá, en suma, cambiar de vaso, algo que se hace más difícil con el correr del tiempo y la experiencia consolidada.

Así, la teoría del derrame de las prácticas docentes apunta a un fortalecimiento de las mismas desde la propia formación inicial. Todo esfuerzo por cambiarlas luego cuando el profesor ya se encuentra, en algunos casos desde hace varios años, implementando estrategias aprendidas durante su formación, podría ser insuficiente para alterar las estructuras con las que ese vaso fue forjado y no lograr reformular el recipiente para el nuevo contenido: el docente, ante la imposibilidad de adaptar su vaso, desecha el nuevo fluido y vuelve a llenarlo con aquel que fue su contenido inicial y que lo tiene bien sabido.

## PROGRAMACIÓN PARA TODOS Y TODAS

La formación docente en tecnología no debería limitarse a uno o dos espacios curriculares dentro del plan de estudios, sino que debería ser un proyecto integral desde estos institutos. Sin embargo, como dijimos, esa fue la actualización tecnológica que se dio en los institutos de formación docente de la CABA.

Más aún, el área de programación se encuentra totalmente ausente de estos currículos. Y en la formación específica del profesorado de Informática, es una materia disciplinar -como Álgebra en el profesorado de Matemática, o Mecánica en el profesorado de Física- desarrollada alrededor de un eje que no es el pedagógico. La vinculación de la programación con los objetivos didácticos de formación del pensamiento computacional, quedan exclusivamente en manos de los cursos de capacitación post título.

Seguramente sea esta una mirada bastante utópica, pero si de verdad queremos integrar la programación al curriculum de la escuela media, haciéndola transversal a todos los espacios curriculares, el primer paso debería ser formar a los futuros docentes en estos conocimientos. Que todos ellos puedan experimentar la lógica de pensamiento computacional para resolver algún problema específico del área que cursan. De este modo, cuando se les presente la posibilidad de realizar un curso de capacitación, los esfuerzos podrán ser puestos en afinar y mejorar esas prácticas, en mostrar nuevos desarrollos, nuevos lenguajes, nuevos dispositivos de software y hardware, y no en cambiarlas.

La lógica que subyace a todo algoritmo es, por exceso de simplicidad, harto difícil de comprender y asimilar. Según el teorema de Dijkstra *todo algoritmo puede ser escrito en base a tres estructuras lógicas de razonamiento: secuencial, selectiva e iterativa* (Dijkstra, Marzo 1968). Esta reducción y simplicidad de la teoría de la programación estructurada, es lo que hace que la resolución de problemas sea tan compleja, ya que se cuenta solo con estas tres herramientas para desarrollar todo tipo de software (exceptuando los que responden a la lógica de la inteligencia artificial, que no es abordada en este trabajo). Por eso, uno de los pilares del pensamiento computacional es la reducción de problemas, o lo que se conoce como técnica *divide and conquer* (Page-Jones, 1988).

Estas pocas bases teóricas alcanzan para elaborar un proyecto que integre la programación a la disciplina particular de estudio. Luego sí, habrá que poner el foco en las técnicas apropiadas para llevar adelante esas bases. Esto último es patrimonio de los cursos de capacitación, ya que las actualizaciones en esta área son constantes; lo primero, lo sólido, lo permanente, debería ser trabajado en los institutos de formación docente a partir de algún proyecto pensado ad hoc. Evitaremos así que los docentes en ejercicio se encuentren con la primera aproximación al pensamiento computacional, recién al realizar su primer curso de capacitación.

## CONCLUSIÓN

Llevar adelante cualquier transformación en el sistema educativo nunca fue tarea fácil. No lo será nunca tampoco. Hay muchos actores involucrados, una fuerte presión de la historia, y, como vimos, un arraigo a ciertas prácticas favorecidas por la trayectoria escolar de los docentes. Tampoco es algo que no demande recursos; por el contrario, grandes partidas presupuestarias deben ser destinadas a este cometido y un enorme conjunto de especialistas, asesores y ejecutores son afectados a la tarea. Por lo tanto, es imprescindible velar para que semejantes esfuerzos no sean en vano. Analizar con cuidado las alternativas, los caminos, las posibilidades, resulta indispensable. Un estudio de tales características, tendiente a la toma de decisiones en este sentido, pertenece al área de la política educativa. No se pretendió aquí ahondar en ese aspecto, sino mostrar que tener en cuenta la formación inicial del profesorado es vital para asegurar el éxito de esos esfuerzos.

Específicamente en el área de la tecnología, y mucho más puntualmente en lo que se refiere a la enseñanza de la programación en las escuelas, el objetivo de formar estudiantes de nivel medio en el desarrollo del pensamiento computacional, no debe ser liderado solo por la capacitación docente



continua, haciendo impactar en ella toda la responsabilidad, sino que es necesario trabajarlo desde la formación docente inicial en orden a evitar las inconsistencias de las prácticas.

Se podría concluir entonces, que un profesor que se ha formado en su disciplina específica, desarrollando proyectos que integran la programación y por ende, el pensamiento computacional, estaría mucho más preparado para llevarlos adelante luego en el ejercicio de su profesión. De este modo, la capacitación docente no tendrá la responsabilidad de “formar” docentes en determinadas prácticas, sino que simplemente “actualizará” los contenidos de un área que ya está formada. Y esto, claramente, es más compatible con los objetivos de tal trayecto formativo.

Este trabajo no pretende ser taxativo ni determinante, sino movilizar una reflexión alrededor de la importancia de la integración de proyectos que incluyan la programación y el desarrollo del pensamiento computacional en los institutos de formación docente.

## BIBLIOGRAFÍA

Adell Segura, J., Llopis Nebot, M., Esteve Mon, F. M., & Valdeolivas Novella, M. (2019). El debate sobre el pensamiento computacional en educación. *RIED. Revista Iberoamericana de Educación a Distancia*, 22(1), 171-186.

Aiello, B., Iriarte, L., & Sassi, V. (8, 9 y 10 de Agosto de 2011). *La narración de la biografía escolar como recurso formativo. VIII Encuentro de Cátedras de Pedagogía de Universidades Nacionales*. Recuperado el 14 de marzo de 2016, de Memoria Académica: [http://www.memoria.fahce.unlp.edu.ar/trab\\_eventos/ev.939/ev.939.pdf](http://www.memoria.fahce.unlp.edu.ar/trab_eventos/ev.939/ev.939.pdf)

Alliaud, A. (2004). La experiencia escolar de maestros "inexpertos". Biografías, trayectorias y práctica profesional. *Revista Iberoamericana de Educación Nro. 34*.

Alliaud, A. (2007). *La biografía escolar en el desempeño de los docentes*. Victoria: Universidad de San Andrés.

Belnicoff, M. A., Consoli, V., & Lungarete, s. (2016). *Educación Tecnológica – Programación. Antecedentes a partir de las normas*. Buenos Aires: Subsecretaría de Planeamiento e Innovación Educativa - CABA.

Branchini, G., Cortez, M. M., Pedemonte, M., & Rodríguez, J. (2019). Las Ciencias de la Computación en el Currículum Escolar. *XXI Workshop de Investigadores en Ciencias de la Computación*. San Juan: Universidad Nacional de San Juan.

Cucuzza, G. (20 de septiembre de 2015). *De #LaInformaticaProhibida a #LaInformaticaComoMateria*. Obtenido de La informática como materia: <http://lainformaticacomomateria.blogspot.com/2015/09/delipalim.html>

Dijkstra, E. W. (Marzo 1968). Go To Statement Considered Harmful. *Communications of the ACM, Vol. 11, No. 3*, 147-148.

Edelstein, G. (2011). *Formar y formarse en la enseñanza*. Buenos Aires: Paidós.

Fair, H. (2010). Hacia una epistemología del neoliberalismo. En *Pensar. Epistemología y Ciencias Sociales*. Rosario: Editorial Acceso Libre.

Feurzeig, W. (1984). *The Logo Lineage*. Obtenido de Digital Deli Archives: [m.edtechpolicy.org](http://m.edtechpolicy.org)

Ley 26206. (2006). *Ley de Educación Nacional*. Buenos Aires.

Martínez López, P. (2019). *Ciencias de la computación para el aula: 1er. ciclo de secundaria*. Buenos Aires: Fundación Sadosky.

Page-Jones, M. (1988). *The Practical Guide to Structured Systems Design*. Londres: Prentice-Hall.

Palacios, A. R. (2011). *La educación del pensamiento a través de la matemática*. Buenos Aires: Lumen.

Parselis, M. (Mayo 2016). El valor de las tecnologías entrañables. *Revista CTS, n° 32, vol. 11*, 71-83.

Program.AR, I. (29 de abril de 2020). *Evaluación de las especializaciones en Didáctica de las Ciencias de la Computación – Resumen ejecutivo del Informe Final 2019*. Obtenido de Program.Ar: <http://program.ar/evaluacion-del-programa-especializaciones-en-didactica-de-las-CC>

Quintanilla, M. Á. (1989). *Tecnología: un enfoque filosófico*. Madrid: Fundesco.

Revelo Sánchez, O., Collazos Ordóñez, C. A., & Jiménez Toledo, J. A. (2018). El trabajo colaborativo como estrategia didáctica para la enseñanza/aprendizaje de la programación: una revisión sistemática de literatura. *TecnoLógicas v.21 n.41*, 115-134.

Sommer, S., Cornejo, M. E., Cortez, M. M., & Rodríguez, J. (2018). El lugar de las Ciencias de la Computación en el curriculum de la escuela secundaria argentina. *XX Workshop de Investigadores en Ciencias de la Computación*. Corrientes: Universidad Nacional del Nordeste.

Stallman, R. (2004). *Software libre para una sociedad libre*. Madrid: Traficantes de sueños.

## “MAPUDUNGUN 2.0”

Prof. Marcia Pizarro  
profemarciapizarro@gmail.com  
Talleres Don Bosco. Zapala, Neuquén

### Resumen

Animarse a incursionar en contextos donde hay respaldos pero poca experiencia o información, nos puede llegar a sorprender para bien, y a hacer propias las palabras, cuando se expresa que podemos comenzar por “Pisos bajos, y llegar a Techos Altos”. Este artículo tiene como finalidad compartir la experiencia de enseñar programación mediante el lenguaje Scratch a estudiantes jóvenes Adultos integrados en la Educación formal Obligatoria, que además de tener discapacidad, pertenecen a Pueblos Originarios.

Esta experiencia es altamente positiva y motivadora, ya que animamos a otros docentes a utilizar esta herramienta, y a ver los contextos desconocidos, como oportunidades de crecimiento y de experiencias de aprendizaje significativas, tanto para el estudiante como para el Docente.

**Palabras claves:** Programación, Discapacidad, Pueblos Originarios

### 1- INTRODUCCIÓN

Podemos decir, que cuando hablamos de “Programación”, inmediatamente piensan aquellos que no la conocen, como una tarea un tanto difícil, que solo está destinada o permitida para un grupo reducido de la sociedad, que, si elige seguir ese camino, tendrá a futuro mucha exigencia intelectual. Y si hablamos de la programación como una tarea ardua, nos parecerá que aún más impedimentos tendrán aquellas personas que posean alguna necesidad especial en su educación.

Otro grupo reducido dentro de nuestra sociedad, son aquellas personas que, desde la intimidad de su hogar, tratan de conservar sus culturas, compartiendo sus enseñanzas y transmitiendo su sabiduría. Luchan para mantener sus costumbres desde su comunidad, sin grandes movimientos políticos.

En mi clase de Computación, se juntaron estas dos realidades: jóvenes con discapacidad, y miembros de comunidades Mapuches activas. El desafío es lograr la conjunción entre estas realidades e incorporar los contenidos que los NAP de Educación Digital, programación y robótica para educación Primaria y Secundaria. En la presentación de los Núcleos, el ministro de Educación, Cultura, Ciencia y Tecnología de la Nación, nos dice:

“El mundo de hoy se caracteriza por cambios y desafíos constantes en donde las nuevas tecnologías nos intervienen como sociedad. Enfrentarlos requiere de la construcción de conocimientos innovadores que permitan resolver los problemas del presente y del futuro. Es por eso que debemos incentivar a todos los chicos de nuestro país para que adquieran las habilidades que requiere el siglo XXI y formarlos para que sean capaces de entender y hacer un uso crítico de las tecnologías digitales en todos los aspectos de la vida y en el mundo del trabajo, en particular”. (Alejandro Finocchiaro, NAP de Educación Digital, p. 3)

### **a- Educación Especial - Discapacidad**

La Ley de Educación Nacional, en su Art. N° 42 establece que:

“La Educación Especial es la modalidad del sistema educativo destinada a asegurar el derecho a la educación de las personas con discapacidades, temporales o permanentes, en todos los niveles y modalidades del Sistema Educativo. La Educación Especial se rige por el principio de inclusión educativa, de acuerdo con el inciso n) del artículo 11 de esta Ley. La Educación Especial brinda atención educativa en todas aquellas problemáticas específicas que no puedan ser abordadas por la educación común. El Ministerio de Educación, Ciencia y Tecnología, en acuerdo con el Consejo Federal de Educación, garantizará por intermedio de diversas estrategias, la inclusión de los/as estudiantes con discapacidades en todos los niveles y modalidades según las posibilidades de cada persona.

El cuidado de la trayectoria escolar de los/as estudiantes con discapacidad implica tomar decisiones con respecto a las estrategias de enseñanza, respetando el principio de no discriminación y el derecho del alumnado a “ser evaluados/as en su desempeño y logros, conforme a criterios rigurosa y científicamente fundados, en todos los niveles, modalidades y orientaciones del sistema e informados/as al respecto” (artículo 126, inciso e).

### **b- Pueblos Originarios**

En la Declaración de las Naciones Unidas sobre los derechos de los pueblos indígenas, en su Artículo 14 nos dice:

1. Los pueblos indígenas tienen derecho a establecer y controlar sus sistemas e instituciones docentes que impartan educación en sus propios idiomas, en consonancia con sus métodos culturales de enseñanza y aprendizaje.
2. Los indígenas, en particular los niños, tienen derecho a todos los niveles y formas de educación del Estado sin discriminación.
3. Los Estados adoptarán medidas eficaces, conjuntamente con los pueblos indígenas, para que las personas indígenas, en particular los niños, incluidos los que viven fuera de sus comunidades, tengan acceso, cuando sea posible, a la educación en su propia cultura y en su propio idioma.

La educación obligatoria en una lengua que los estudiantes no entienden o manejan es un obstáculo para que se disfrute su derecho a la educación. Sin embargo, esto no quiere decir que tenga que recibir información solamente en su lengua materna.

No conocer las lenguas oficiales puede ser también un obstáculo para su desarrollo y su integración en la sociedad en la que la persona vive.

La Asamblea General de la ONU afirmó que la protección de las lenguas indígenas implica preservar y promover la cultura de respeto al medio ambiente de los pueblos que las hablan, una condición indispensable para el desarrollo sostenible.

Evo Morales, ex presidente de Bolivia, durante una de las Asambleas, expresó:

“Ejercer la lengua es uno de los derechos más importantes para los pueblos indígenas y, por lo tanto, para la humanidad. La lengua es cultura y proyecta una cosmovisión, una forma de ver el mundo, de vincularse con otros (...) Si desaparecen las lenguas, desaparecerán las memorias y cultura de los pueblos.”

### c- Educación Digital, inclusiva y accesible

La producción de información es un derecho de todos los ciudadanos, por este motivo existen pautas de accesibilidad que permiten que usuarios con diferentes características tengan las mismas oportunidades de crear, independientemente de sus capacidades técnicas, cognitivas o físicas.

Refiriéndose al acceso del aprendizaje, La Declaración de Nueva Delhi (2015) en el inciso 6 de su preámbulo afirma: “[...] el acceso universal a la información y el conocimiento, mediante las tecnologías de la información y la comunicación (TIC) y las tecnologías auxiliares, en igualdad de condiciones con los demás, es para las personas con discapacidad un derecho humano inalienable y una condición previa para vivir de forma independiente y participar plenamente y en pie de igualdad en la sociedad. (p. 3)”

Esta idea se refuerza en los Objetivos de Desarrollo Sostenible (ONU – ODS 2015), que nos dice en su objetivo n.º 4: “... se centra en Garantizar una educación inclusiva, equitativa y de calidad y promover oportunidades de aprendizaje durante toda la vida para todos”.

Su meta 4.5 detalla: “De aquí a 2030, eliminar las disparidades de género en la educación y asegurar el acceso igualitario a todos los niveles de la enseñanza y la formación profesional para las personas vulnerables, **incluidas las personas con discapacidad, los pueblos indígenas** y los niños en situaciones de vulnerabilidad (2015)”.

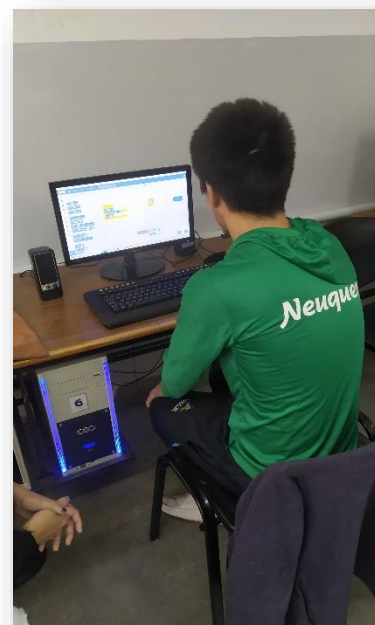
## 2- CONTEXTO

En el contexto de una educación inclusiva existen diferentes recursos y configuraciones de apoyo para alcanzar la accesibilidad. Su implementación en las instituciones educativas debe realizarse en función de las barreras obstaculizadoras del aprendizaje que se identifiquen.

Por otro lado, La educación inclusiva exige hoy en día el manejo de todos aquellos recursos que puedan ayudar a promover la multialfabetización y la inculturalidad para favorecer el acceso al aprendizaje de todos los estudiantes.

La inclusión implica un enfoque aglutinador de aspectos filosóficos, sociales, políticos, económicos y, especialmente, pedagógicos.

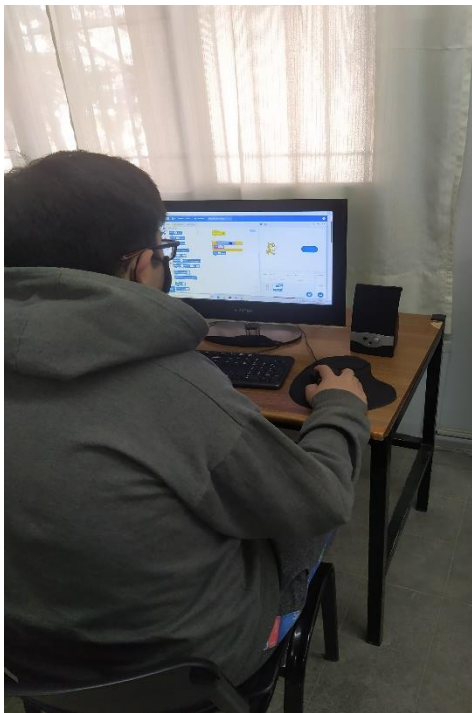
*De este modo, es que comenzamos a abordar este proyecto: mediante una educación Inclusiva y Digital. Buscamos el aprendizaje constructivo e integral, valorando capacidades*



*de todos los estudiantes; respetando sus propios orígenes y culturas para que todos se sientan partícipes.*

### 3- DESARROLLO DE LA EXPERIENCIA

Esta experiencia se desarrolla en un grupo de 4 estudiantes con características muy diferentes. Si resumimos su trayectoria en la educación formal, debemos indicar que son estudiantes que estuvieron en la escuela Especial, y hoy transcurren el ciclo de primaria para jóvenes adultos.



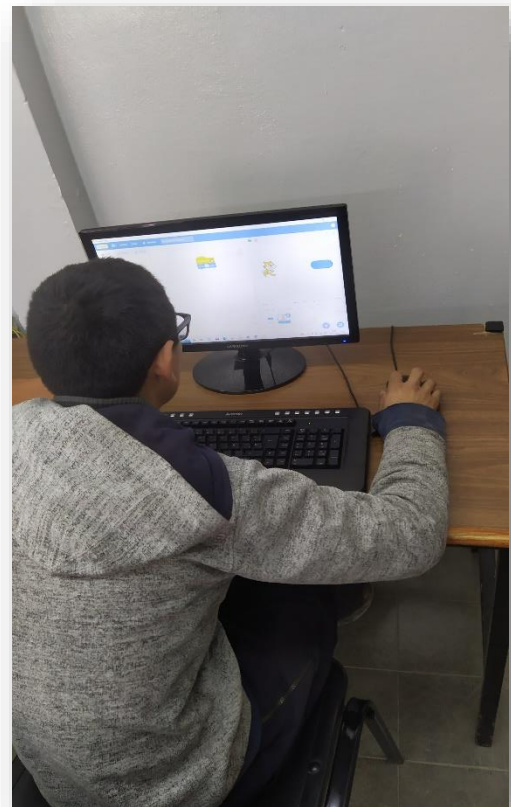
Sus edades rondan entre los 17 y 21 años, pero siguen teniendo corazones de niños. Sus características son muy variadas y en algunas situaciones sus capacidades son asombrosas. Para destacar sus cualidades, además de ser buenos programadores, tengo un deportista que compite en Atletismo, y participa de todos los torneos que lo invitan. Además, uno de ellos se destaca en su capacidad de estudio: es evidente que va a poder transcurrir y seguramente aprobar una formación secundaria. También tenemos a un trabajador incansable, siempre dispuesto a ayudar a su papá, sin tener en cuenta sus dificultades. Y por último tengo un romántico, galán, que comenzó a andar de la mano de Cupido, y a enamorar a cuanta señorita se cruza por su camino.

Comenzamos durante el año 2019. La característica principal del proyecto

era trabajar en “realidad Aumentada” utilizando una cámara Web. Llegamos a realizar el juego, pero por pandemia, no pudimos dar continuidad al proyecto, y logramos re iniciarlo en el presente año 2022 con nuevas ideas.

**El objetivo principal** de todo el proyecto siempre fué la introducción al pensamiento computacional, mediante la programación con Scrath, demostrando que la Tecnología, y en especial la programación, no son barreras para nadie, y de este modo fomentar la conciencia social para la inclusión y la autonomía de este grupo.

En cuanto al **contenido** Inicial, si bien fue un proyecto “dinámico” en donde vamos evaluando las capacidades de cada uno, y hasta “donde podemos llegar”, se logró trabajar dos conceptos puntuales de programación:

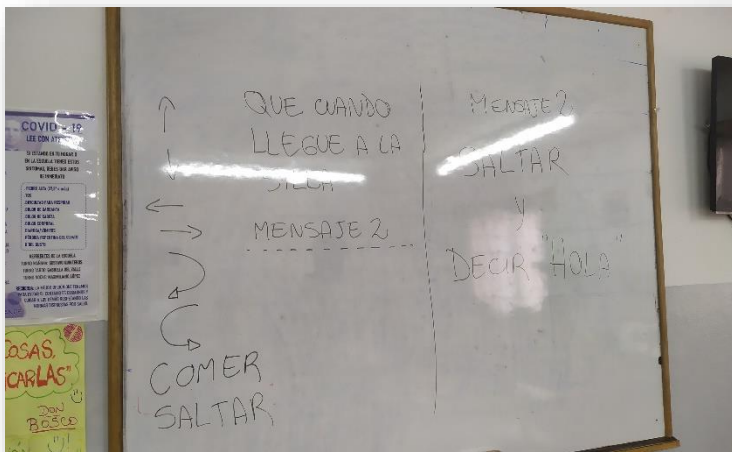
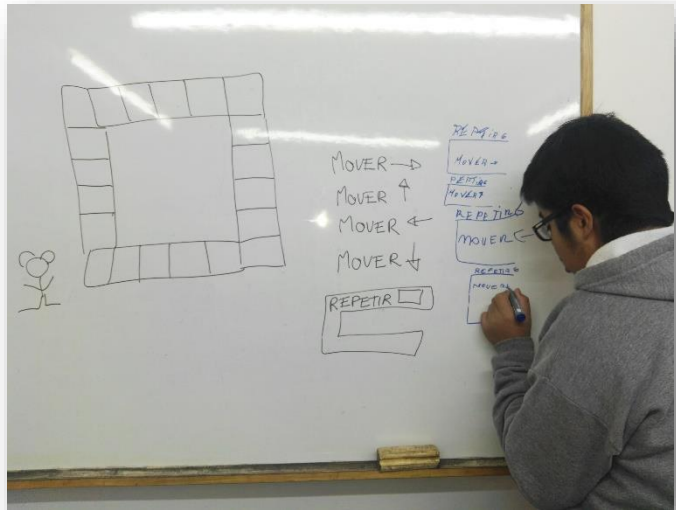




**Repetición:** donde los chicos aprenden a reconocer patrones, es decir “lo que es igual”, para elaborar instrucciones que permitan resolver repetidamente este problema sin necesidad de que el programa sea larguísimo.

**Alternativa condicional:** es comprender una estructura del lenguaje de programación que consta de dos partes: una condición y una acción; la acción se realiza solo cuando se verifica la condición. Esto nos permite construir programas que tomen decisiones y cuyo comportamiento no sea siempre el mismo

En cada clase, debemos recordar, casi desde



cero, todos los comandos que vamos aprendiendo. Siempre jugamos a ser “autómatas” o “robots” que reciben indicaciones. En el aula, armamos diferentes escenarios, con dificultades, y cada uno, por turno, va indicando el camino para resolver el problema. Luego, reforzamos lo aprendido en el pizarrón, y volvemos a escribir cada indicación. Por último, estas indicaciones las asociamos a los bloques que son los comandos que armarán nuestro algoritmo.

## Herramientas y Dificultades

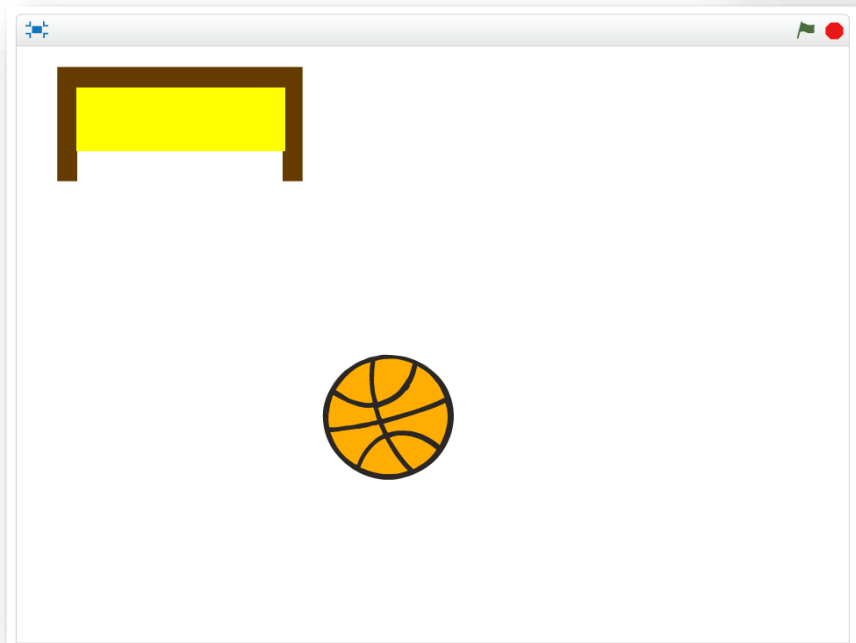
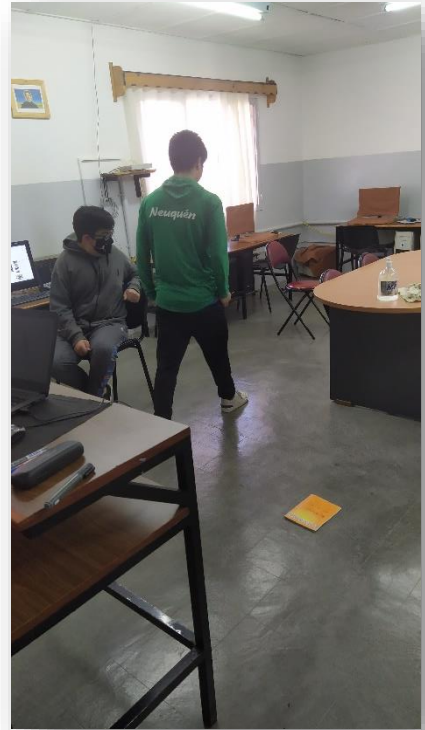
Para conocer la programación por medio de bloques, iniciamos con Pilas Bloques. Allí comenzaron a demostrar la capacidad para adquirir el conocimiento, y el entusiasmo que se necesita para ir logrando cada desafío. Se mostraban muy orgullosos de sus logros al ir avanzando de nivel.

Luego, al pasar al Scratch comenzamos a trabajar en la creación de un propio juego o aplicación. En esas instancias, descubrí que dos de los estudiantes comenzaban a tener dificultades para la lectura de los comandos: les cuesta distinguir la letra mayúscula o minúscula, y la letra de imprenta o cursiva. Este inconveniente no existía en Pilas Bloques, porque al lado de cada bloque hay un dibujo que indica lo que hace ese bloque (ejemplo: flechas →). Subsamos esa dificultad tomándonos el tiempo en la lectura de cada comando y asociándolo con los colores.

**Quiero destacar que, a pesar de tener el inconveniente de la lectura, esto no coartó el avance en adquirir los conceptos de programación.**

Tuvimos dos inconvenientes más, que lejos de limitarnos, fueron el puntapié para desarrollar las dos etapas del proyecto: al estudiante “A” le comenzó a costar incorporar los conceptos de repetición, por tanto, comenzó a costar el avance en la incorporación de contenidos, y al estudiante “B” le comenzó a aburrir esto de la “programación”, no había nada que lo atrae o entretenga.

Quisimos que el estudiante “A” que no pudo adquirir los conceptos de repetición, se sintiera parte del proyecto, por esto, decidimos hacer un juego de “Realidad Aumentada”, que consiste en dos jugadores: un Jugador frente a la pantalla que dá las indicaciones al 2do jugador que está frente a la cámara. Estas indicaciones son movimientos que tiene que hacer para que un objeto, “una pelota” se mueva, entre en un arco y hagan un “gol”. Internamente, en el algoritmo, se programa con bloques sensores que captan los movimientos de una cámara encendida. Hay una condición y una repetición, en donde el objeto se mueve hasta que toque el color del Arco. Entonces ahí muestra: “Gol..!”. El estudiante se integra al momento de hacer los movimientos o dar las indicaciones. No conoce en profundidad lo que internamente hace el juego, pero sí sabe, y muy bien, como jugarlo. De esta forma se siente y es parte activo del proyecto. Este fue el “inconveniente” que dio como resultado la primera parte del proyecto.





Con respecto al estudiante “B”, dialogamos mucho y fuimos conociendo los temas que realmente le interesan. Así llegamos a charlar de su familia, y sus orígenes: El y su familia pertenecen a la comunidad Mapuche. Se llena de entusiasmo al contar sus costumbres, sus celebraciones. Cuenta al detalle cada uno de los acontecimientos que vive en su comunidad. Este fue el punto inicial para poder conectar con El, y dar inicio a la segunda edición de nuestro proyecto.

Mediante una nueva estrategia para favorecer el aprendizaje del estudiante: incorporamos las costumbres y la Lengua Mapuche al juego que programamos.

## Resultados

Utilizando los mismos conceptos de programación (Repetición simple y Condicional) modelamos nuestro juego: ya no era una “pelota que ingrese al arco” sino una “ovejita o chivito que ingresa a su corral” simulando “la Marcación o Señalada”. También incorporamos la enseñanza de la lengua Mapuche: con la aplicación aprendemos a decir y a escribir los números, los colores, algunas frases comunes en Mapudungun, y conocemos algunas de las celebraciones del Año Mapuche.

Esto hizo que mi rol como docente, cambie al rol de “aprendiz” junto a 3 de mis 4 estudiantes. Pasamos horas escuchando a mi Estudiante “B”, tomando nota e imaginando como plasmar su cultura y su lengua en nuestro juego. También participa la mamá del estudiante, porque es ella la que nos enseña la lengua, y su pronunciación.

Al comenzar a interactuar con los miembros de la comunidad, y descubrir el cuidado y reserva que tienen de sus costumbres, necesité acercarme al Lonko de la comunidad, y comunicarle el trabajo que estaba realizando. Es una experiencia más que satisfactoria, ya que no solo recibí la aceptación y agradecimiento por difundir su lengua y su cultura, sino también me invitaron a participar de una de sus ceremonias más importantes: El nguillatun: es una rogativa, la más importante para el pueblo Mapuche. En él, agradecen y piden por el bienestar de la familia, la comunidad, las tierras y cosechas. Muchas veces y en clase, comentamos esta celebración, pero terminé de maravillarme y entenderla cuando pude presenciarla. La solemnidad de la ceremonia y el respeto a lo sagrado es palpable. Cada momento tiene su significado y está llena de símbolos.

Al momento de escribir este artículo, está por iniciar la celebración del “We Txipantu”, es el año nuevo Mapuche, que implica la renovación de las personas en su espíritu, es la noche más larga del año, donde al amanecer se espera la nueva salida del Sol, con el cuerpo y el espíritu limpio y sintiendo la fuerza del Futachao o Padre Grande.

También, buscando material, pude conectarme con profesores de lengua Mapuche del vecino país de Chile. Compartir saberes, fue más que gratificante.



*Así, nos involucramos de forma activa con todos los estudiantes, y los entusiasmos con experiencias significativas relacionadas a su contexto sociocultural y a sus intereses, promoviendo el aprendizaje significativo. A la vez que se convierten en productores de conocimiento y no en consumidores pasivos de saberes.*

#### **4- CONCLUSIÓN**

Son muchas las satisfacciones. Para remarcar algunas:

Incentivamos la creación de programas en los estudiantes, como una posibilidad para todos y no solo para algunos, descubriendo que nuestra condición no debe ser nunca un impedimento para concretar lo que soñamos.

Estimulamos la confianza en ellos, y su potencial, mediante el uso y la ejecución de programas diseñados por ellos mismos.

Promovimos la reflexión crítica y el trabajo colaborativo, a través de la detección de dificultades en el aprendizaje del grupo y de errores propios en los algoritmos.

Despertamos en los estudiantes el interés por sus orígenes y costumbres, y la generosidad de compartirlos en aquellos que lo desconocen.

#### **5- PROPUESTAS A FUTURO**

Como trabajo futuro se propone ir actualizando la aplicación, mediante las necesidades de aprendizaje de la cultura Mapuche que vayan surgiendo, respetando los conceptos de programación aprendidos y así lograr dar solución.

Además, estamos en condiciones de compartir la aplicación a otros pueblos Originarios, para que puedan enseñar y difundir sus culturas mediante nuestro juego. Al ser programado en una plataforma de código abierto, podemos mostrar internamente como es el algoritmo del juego, y escribir “manuales” que indiquen, de forma sencilla, como modificarlo y hacerlo propio.

También, tengo la posibilidad de compartir este trabajo con estudiantes del Nivel Secundario, y convertirlo en una App para Celulares.

*Al intercambiar y compartir nuestras experiencias pedagógicas, también producimos saberes, y expandimos el aula, más allá de la escuela. Compartir nuestras prácticas innovadoras, nos permite construir en Red.*

**Prof. Marcia Pizarro**

## Referencias

Núcleos de Aprendizajes Prioritarios (NAP) de Educación Digital, Programación y Robótica (2004 – 2012). Consejo Federal de Educación. Orientaciones para la elaboración de material digital accesible (2015). Programa Aprender Conectados. Ministerio de Educación, Ciencia y Tecnología de la Nación

Declaración de Nueva Delhi sobre TIC inclusivas al servicio de las personas con discapacidad: Hacer del empoderamiento una realidad (ONU, 2015). Recuperado de: <http://unesdoc.unesco.org/images/0023/002320/232026s.pdf>

Objetivos de Desarrollo Sostenible (ODS) (Naciones Unidas, 2015). Recuperado de: <http://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>

Preservar las lenguas indígenas es preservar la herencia común de la humanidad. (2019) Fuente: Noticias ONU <https://www.un.org/es/desa/indigenous-languages>

Recuperar y fortalecer las lenguas indígenas es una necesidad apremiante para el futuro de la humanidad. (2019). Fuente: Noticias ONU <https://www.un.org/es/desa/international-year-of-indigenous-languages-2019>

## El ABP en el redictado de materias de programación: una experiencia en la educación superior

Natalia Colussi<sup>1</sup>, Natalia Monjelat<sup>2</sup>, Pamela Viale<sup>3</sup>

<sup>1</sup> Licenciada, Facultad de Ciencias Exactas, Ingeniería y agrimensura (UNR), Rosario, Argentina

<sup>2</sup> PhD, Instituto Rosario de Investigaciones en Ciencias de la Educación (IRICE, CONICET-UNR),  
Rosario, Argentina

<sup>3</sup> PhD, Facultad de Ciencias Exactas, Ingeniería y agrimensura (UNR), Facultad de Química e  
Ingeniería del Rosario (UCA), Rosario, Argentina

[colussi@fceia.unr.edu.ar](mailto:colussi@fceia.unr.edu.ar), [monjelat@irice-conicet.gov.ar](mailto:monjelat@irice-conicet.gov.ar), [pamela@fceia.unr.edu.ar](mailto:pamela@fceia.unr.edu.ar)

### Resumen

En este trabajo se describe y analiza una experiencia de enseñanza de la programación, un contenido clave de las Ciencias de la Computación (CC), desarrollada durante los últimos 5 años en el segundo dictado de las materias Programación I y Programación, para la Lic. en Ciencias de la Computación, la Lic. en Matemática y el Profesorado en Matemática de una universidad pública argentina. El diseño de la experiencia parte de los principios del Aprendizaje Basado en Problemas y Proyectos, que ha implicado una redefinición de las actividades didácticas a desarrollar en la cursada. A partir de las experiencias implementadas, es posible señalar múltiples aprendizajes y fortalezas de este enfoque tales como la posibilidad de seguimiento del progreso en el aprendizaje, la rápida adaptación al cambio de contexto presencial-virtual, la disminución del plagio al tratarse de producciones originales y creativas, entre otras. Estos resultados ponen en valor la implementación del ABP para la enseñanza de conceptos de CC, particularmente en este caso, desde contenidos disciplinares de programación y pensamiento computacional. Se plantea también la necesidad de revisar las prácticas docentes universitarias, dando lugar a innovaciones didácticas que desde perspectivas activas, habiliten aprendizajes significativos en el estudiantado.

**Palabras clave:** Aprendizaje basado en proyectos y problemas, enseñanza y aprendizaje de la programación, redictado, pensamiento computacional, ciclo inicial de la educación superior.

### Introducción

El Aprendizaje Basado en Proyectos y Problemas (ABP) es un modelo de aprendizaje y de enseñanza que surge, para los *problemas* en los cursos de química y medicina de la Universidad de McMaster en Canadá a finales de los años 70, y para los *proyectos* en el Centro Universitario Roskilde y en la Universidad de Aalborg ambos en Dinamarca, a principios de los 70 (Gómez, 2005; García Martín y Perez Martínez, 2018). En ambos casos, tanto para el aprendizaje basado en problemas como para el aprendizaje basado en proyectos se observa como aspecto común, el hecho de considerar al estudiantado como protagonista del proceso de enseñanza-aprendizaje, quienes construyen el conocimiento al mismo tiempo que lo aplican en situaciones que se asemejan a las que se enfrentarán en la vida profesional cuando tenga que abordar problemáticas concretas del

área. Desde esta perspectiva, el docente reconvierte su rol para transformarse en un guía de los estudiantes en el desarrollo de aprendizaje, orientándolos hacia su progreso en la construcción de saberes por medio de una continua evaluación de todo el proceso (Domínguez et al., 2008; Heydrich et al., 2010).

La implementación de esta metodología de enseñanza y aprendizaje en los primeros años de la universidad, permite observar claramente cómo los estudiantes progresan en la adquisición de conocimiento y saberes disciplinares, ya que conlleva un seguimiento mayor por parte de los docentes (García Martín y Perez Martínez, 2018). Este acercamiento provoca que los docentes y estudiantes logren vincularse de una forma más próxima, haciendo que el seguimiento del proyecto por el cual se canaliza el aprendizaje, permita una observación mucho más clara de cómo se logra la comprensión de los temas a través del diseño de la solución (modelo+estrategia+código) que los estudiantes producen como soluciones parciales de un proyecto.

En este artículo, se describe una experiencia de redictado desarrollada en la Universidad Nacional de Rosario, recurriendo a los aportes del Aprendizaje Basado en Proyectos y Problemas (ABP) (Domínguez et al., 2008; García Martín y Perez Martínez, 2018; Sanchez y Blanco, 2012; Queiruga-Dios et al., 2019). Se buscó responder a las múltiples problemáticas subyacentes de los cursantes del redictado mediante la realización de dos proyectos grupales de programación, en el marco de un “Plan de Trabajo Didáctico para el Aula” (PTDA) (Colussi y Viale, 2019; 2020), que implicó la re-elaboración tanto de la presentación de los contenidos, la ejercitación y la modalidad de evaluación.

## **Descripción de la experiencia**

### **Contexto y Destinatarios**

Aplicamos la práctica del ABP en las cátedras de Programación I y Programación en su segundo dictado, buscando una abordaje de los contenidos de una forma innovadora centrada en los estudiantes pero asumiendo que no se requiere ningún conocimiento previo de los alumnos. Éstas materias corresponden al primer año de la Licenciatura en Ciencias de la Computación (LCC) y segundo año de la Licenciatura y Profesorado de Matemática (LM y PM). La mayoría de los estudiantes del redictado son recursantes del primer cuatrimestre, aunque también es posible que se incorporen alumnos que nunca la han cursado. No existe requisito alguno para inscribirse en el redictado. Aquellos estudiantes que ya la cursaron buscan en su segunda cursada una oportunidad de aprendizaje, tanto de contenidos, como de saberes, habilidades y competencias que no fueron alcanzados en el primer intento. Aunque existen heterogeneidades en el curso, es posible identificar una serie de problemas en el estudiantado que pueden agruparse en: i) la no adaptación al cursado y las metodologías de evaluación, ii) a la dificultad en la incorporación de nuevos conceptos, herramientas y el paradigma de programación, iii) a la no integración a la comunidad académica y espacios educativos, el entorno social entre pares, la relación con la comunidad docente y científica, como así también la adaptación a la vida universitaria. En lo referido a cuestiones técnicas disciplinares, los estudiantes en su mayoría presentan dificultades para: a) formular un problema y diseñar soluciones algorítmicas, b) descomponer un problema en subproblemas, c) representar los

datos, d) modelar, analizar y organizar los datos de forma abstracta, e) generalizar, generar y automatizar casos de prueba, etc.

La **tabla 1**, muestra de forma resumida, la cantidad de inscriptos, cursantes y los grupos conformados cada año, junto a la cantidad de integrantes que los componían en promedio durante la ejecución del ABP en el aula. Cabe destacar, que a partir del año 2019 se incorporan al redictado los cursantes de las carreras LM y PM, que otros años no tenían en su currícula la materia Programación. A su vez, a partir del año 2021, la población de estudiantes que ingresan a la carrera LCC varía significativamente respecto de otros años, provocando que los cursos se tornen masivos tanto en el primer dictado como en el segundo.

Año de Cursada	Cantidad de Inscriptos	Cantidad de Cursantes	Cantidad de Grupos	Cantidad de Integrantes por Grupo	Cantidad de Docentes
2017	20	17	8	2	2
2018	21	21	8	2 y 3	2
2019	21	21	5	3	2
2020	26	26	10	2 y 3	2
2021	90	60	26	4 y 3	5

**Tabla 1.** Resumen de la aplicación de ABP entre 2017 y 2021 respecto a cantidad de Estudiantes, Grupos y Docentes del Redictado

## Contenidos

El curso se estructura en base a dos proyectos. La realización de los mismos ejercita a los estudiantes sobre temas disciplinares fundamentales de la programación (Felleisen et al., 2001; Bloch, 2010; Martinez Lopez, 2013). La **figura 2** detalla qué contenidos son abordados en el primer y segundo proyecto, así como aquellos transversales que se contemplan en ambas instancias. Si bien estos dos desarrollos constituyen el eje central de la materia, la misma se complementa con el dictado de clases de práctica y teoría y parciales para complementar la evaluación. A fin de explicitar un ejemplo del tipo de proyecto que se realiza, se menciona la propuesta del año 2021, donde *el primer proyecto* consistió en desarrollar un programa que produjera una serie de androids y posteriormente lograra transformarlos entre sí bajo el efecto del artístico de Andy Warhol. Los estudiantes tuvieron la libertad de elegir entre distintos personajes para diseñar y transformar. Se trabajó sobre objetos gráficos diferentes, pero que compartían muchas de sus características, lo que permitió la detección temprana de patrones entre los distintos diseños seleccionados. Con la guía de las docentes, fueron capaces de ir proponiendo abstracciones cada vez más generales e implementarlas a través de funciones parametrizables. A medida que avanzaron en el desarrollo de los proyectos, se fue observando cómo se asimilaron las ventajas de trabajar de forma modular. En *el segundo proyecto* se trabajó sobre animaciones, introduciendo el paradigma de programación orientado a eventos. Cada grupo eligió un tipo de animación a desarrollar, utilizando los androids del primer proyecto como personajes o parte de la animación. La idea aquí fue mostrar que muchas veces desarrollamos código con un fin y luego debemos adaptarlo o reutilizarlo en un contexto

diferente al que habíamos considerado en un principio. El desarrollo de ambos proyectos constituyó un verdadero ejemplo de un trabajo de forma incremental.

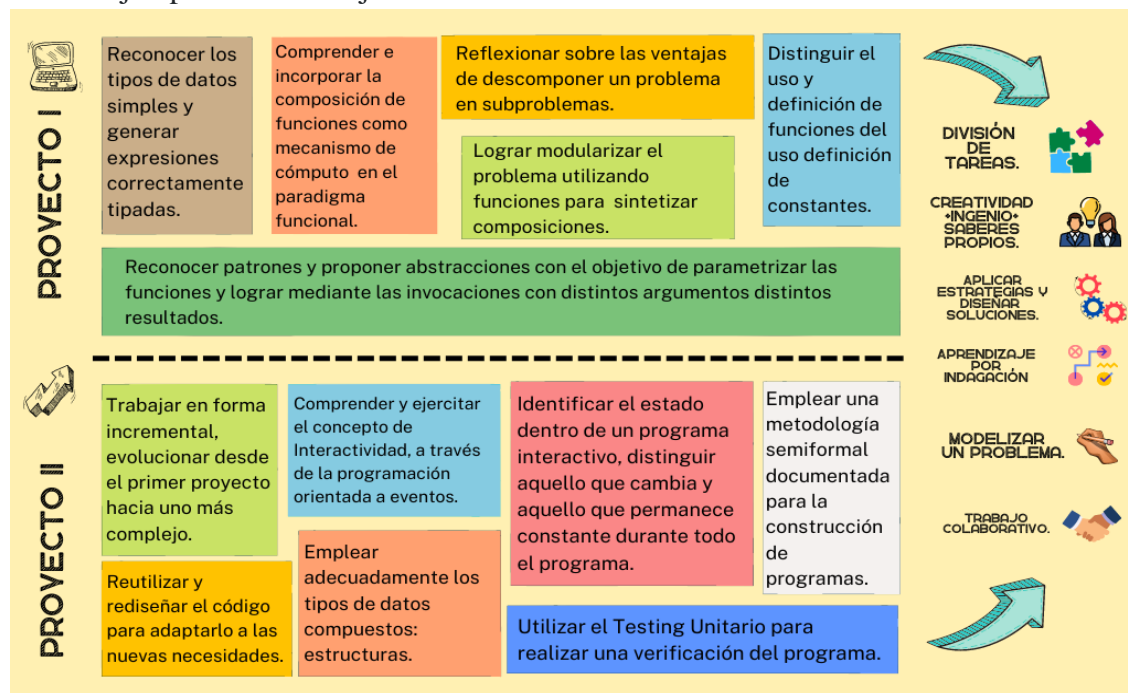


Figura 2. Objetivos Disciplinarios Perseguidos por Proyecto, en nuestra experiencia de ABP

## Metodología

La implementación del ABP en el aula del redictado de programación se lleva adelante en las siguientes etapas o fases:

### 1. Selección de los grupos de trabajo (1er semana):

Los estudiantes conforman grupos de trabajo durante la primera semana de clases, de dos o máximo tres integrantes, o en el caso de que el curso fuese masivo (más de ochenta cursantes) se divide el mismo en grupos de tres o cuatro integrantes.

### 2. Presentación del trabajo por proyectos en el aula y detalles del problema a trabajar (1er semana):

Al comenzar el cursado se les explica a los estudiantes que se trabajará en grupos en dos proyectos de programación a lo largo de la cursada. Se establecen los hitos más importantes dentro de cada uno de los proyectos. Se detalla un problema específico, con ciertos lineamientos y un contexto que está sujeto a una temática, pero abierto a múltiples modificaciones y variantes, de las cuales se desprenden los diferentes proyectos grupales. Cada proyecto preserva ese origen común y año a año la temática de los proyectos distingue a la cursada. Para el caso de Proyecto I corresponde a la primera semana de clases y para el Proyecto II, el inicio corresponde a la novena semana de clases.

### 3. Desarrollo y Revisión de los proyectos (2da, 3era, 4ta, 5ta y 6ta semana):

La forma de resolución de los proyectos es incremental, metodológica y supervisada por los docentes mediante las instancias de revisión semanales, usualmente denominadas controles, donde por turnos asignados a cada grupo se trabaja sobre la revisión y evaluación de cada instancia puntual de los proyectos. La semana posterior a la presentación los estudiantes,



por grupo, presentan el anteproyecto de su trabajo. Este constituye el primer control sobre el trabajo que realizarán. En esta etapa se fomenta el trabajo autónomo del grupo, pueden utilizar toda la bibliografía de la materia, las ayudas que brinda el entorno de programación para buscar funcionalidades ya definidas o predefinidas en el lenguaje y hacer uso de ellas en su proyecto. Se destinan horas de clases para trabajar sobre los proyectos y también se utilizan horas dedicadas específicas al mismo fuera de clase del mismo modo que se haría con la realización de una práctica convencional. Para el Proyecto I, se estima un total de siete/ocho semanas de desarrollo y para el Proyecto II un total de seis/siete semanas.

**4. Evaluación y Entrega de los proyectos (7ma semana):**

La evaluación de la materia contempla un proceso formativo y continuo logrado a partir del seguimiento semanal de los proyectos donde mediante un registro de cada encuentro en una planilla de cálculo donde se evalúa el trabajo realizado por cada grupo. La entrega del trabajo se realiza mediante la plataforma moodle donde se aloja el sitio de la materia. Los grupos entregan los archivos con los códigos producidos y la documentación solicitada la cual contempla la realización de una encuesta.

**5. Defensa y Socialización de aprendizajes (8va semana):**

La defensa de los proyectos se desarrolla también frente a los pares, permite a los alumnos presentar el fruto de sus esfuerzos, pero también apreciar críticamente el trabajo de sus compañeros, aprendiendo de ambas experiencias. Se busca que aparezca por comparación una crítica constructiva o valoración a la calidad del trabajo propio y el de los otros grupos, estableciendo con ello pautas para mejorar ambos trabajos. Es por ello que dicha instancia implica un escenario sumamente valioso dentro de la propuesta didáctica, enriquecedor para todos los cursantes y docentes.

## Herramientas utilizadas

Las herramientas empleadas en la cátedra se agrupan en las siguientes categorías:

- **Construcción de Programas y aplicación de conceptos del Pensamiento Computacional:** se utiliza el Lenguaje funcional, Racket (<https://racket-lang.org/>) y su entorno de desarrollo, Dr. Racket, una herramienta diseñada especialmente para la enseñanza de la programación, concebida con un fin pedagógico, desarrollada por Matthias Felleisen y colegas desde el año 1994 hasta la fecha, basada en el lenguaje Scheme.
- **Control o Tutoría de Proyectos:** Se emplean planillas de cálculo compartidas, google sheets, para sistematizar el proceso de seguimiento y evaluación formativa. Las planillas reúnen indicadores vinculados al pensamiento computacional, para registrar y analizar la evolución de los proyectos. En la virtualidad y las clases híbridas se utiliza la plataforma google meet para conectarse con los estudiantes, grabar y luego compartir las devoluciones para su posterior revisión y repaso. Se emplea la plataforma moodle asociada a la cátedra para subir las grabaciones y anunciar los horarios de control de cada grupo.
- **Entrega de Proyectos y Comunicación con y entre los estudiantes:** Se utiliza la app Telegram para comunicarnos con los estudiantes, mediante un grupo creado para tal fin, el cual permite estar en contacto de una forma rápida. Se usa la plataforma moodle para la entrega de los anteproyectos, proyectos, videos y cuestionarios finales.



- **Desarrollo del Espacio Virtual de Exposición de Proyectos:** Se utilizó la plataforma google sites para crear el sitio web que contiene: a) los videos grupales con las exposiciones de los proyectos; b) el espacio de intercambio asíncrono entre los grupos de estudiantes, donde se empleó para tal fin, formularios y planillas de cálculo de google, que permitieron dejar los comentarios y visualizar cada video al mismo tiempo.

## Resultados

En este apartado señalaremos algunos resultados destacados de la experiencia desarrollada durante los últimos cinco años, focalizando en los aprendizajes logrados como cátedra luego de las diferentes implementaciones y en las fortalezas detectadas a partir de las mismas.

En cuanto a los aprendizajes ganados, las experiencias desarrolladas ratifican que en la implementación del ABP el trabajo grupal es un pilar fundamental en la concreción de los proyectos. Esta forma de trabajo sostiene el trabajo y motiva a finalizarlo, brindando apoyo y contención entre pares tanto en la adquisición conjunta de saberes y competencias como permitiendo la ejercitación de metodologías de trabajo vinculadas a las formas profesionales del trabajo de un programador. En este contexto, el rol del docente consiste en supervisar y guiar el trabajo grupal, buscando que surjan los liderazgos y los roles de cada integrante, en una actitud activa y participativa de todos sus miembros. Desde ese rol, la división en grupos se deja librada a los estudiantes, aunque de ser necesario, se interviene reagrupando a los alumnos en el caso que no hallen por sí mismos compañeros de grupo, o en el caso de que manifiesten algún inconveniente entre ellos o ante la deserción de la mayoría de sus integrantes. En relación con el trabajo grupal, se ha identificado que conservar los grupos de trabajo en los dos proyectos ayuda al seguimiento y evaluación cuatrimestral de los estudiantes, facilitando la integración y el desarrollo incremental y composicional de ambas propuestas, al mismo tiempo que se aprovecha los vínculos ya establecidos y la dinámica alcanzada en el funcionamiento del grupo.

Por otro lado, otro aprendizaje que ha dejado la experiencia es la importancia de acompañar la presentación de los proyectos de diferentes formas durante su desarrollo. Por ello, al inicio de la cursada se presenta una guía explicativa, donde se trazan los objetivos a cumplir en el corto plazo, semana a semana. Ese acompañamiento luego los involucra mediante la realización de un anteproyecto y en la asistencia al primer control del trabajo se discute con ellos la factibilidad de la realización del mismo en el plazo estipulado, corrigiendo aquello que por exceso o por defecto se haya pensado como parte del Proyecto I. En el caso del Proyecto II es importante el trabajo incremental y darle otra dimensión a la propuesta del Proyecto I, para que forme parte de una animación, juego, o simulación que se programará bajo la misma metodología de trabajo e involucrando otros conceptos que se abordan durante el cursado como la programación orientada a eventos. Se discute con cada grupo animaciones y variantes de simulaciones típicas (juego de la memoria, triple jack, etc.) previamente pensadas por los docentes, aunque cada grupo tiene la libertad de presentar una idea propia, un deseo de programa, animación o juego que se anhele realizar, debe mediar una inquietud por parte de los estudiantes hacia el problema, un interés resolutivo hacia el mismo, y se los acompaña para que esto finalmente ocurra. Este caso constituye un desafío propio y particular para los estudiantes, y de igual manera para los docentes, quienes piensan junto a los estudiantes la resolución algorítmica y se encuentran impulsados también por la misma energía y motivación de la búsqueda de la construcción del programa que finalmente

resuelva el problema planteado. Cada proyecto resulta entonces enriquecedor y un espacio de aprendizaje para todos: estudiantes y docentes.

En línea con lo anterior, el seguimiento a cada grupo constituye uno de los ejes centrales del éxito en la concreción de los proyectos, dando en cada instancia, en cada encuentro, una devolución sobre lo realizado junto a un conjunto de metas a cumplir para el próximo encuentro. Cada una de estas citas grupales duran unos 15 min o 20 min. Se establece una dinámica de exposición corta, donde ellos nos cuentan qué avances lograron concretar, qué dificultades surgieron, si tienen algún problema adicional también lo manifiestan, se genera un vínculo de confianza entre los estudiantes y docentes que permite liberar sinceramente las dudas y a su vez nos permite ver qué problemas potenciales pueden llegar a presentarse si en el grupo los liderazgos no surgen o si estos abruma, o si por el contrario no logran congeniar y amalgamar las ideas y trabajos. Los estudiantes pueden luego, si quedan aún dudas por consultar, solicitar trabajar sobre el proyecto en los horarios de consulta de la materia, o juntarse a trabajar sobre el proyecto durante las horas de práctica, usando los espacios de los laboratorios en la universidad como lugar de encuentro y trabajo grupal. Es importante que estos espacios también existan fundamentalmente para los estudiantes con mayores inseguridades.

Cabe destacar también lo valioso que resulta la difusión pública del trabajo realizado. La fase final del ABP, generalmente implica no sólo la exposición en el aula, sino también un espacio de divulgación, el cual se consigue mediante la exposición de los proyectos en las “Jornadas de Ciencias de la Computación”<sup>1</sup> (JCC). Este evento abierto al público académico en general, constituye un cierre armónico del proceso de aprendizaje llevado adelante durante la realización del proyecto. Esta instancia se produce unas dos o tres semanas posteriores a la defensa del Proyecto I, por lo cual, los estudiantes logran además corregir o mejorar sus proyectos a partir de las devoluciones dadas en la defensa, llegando a las JCC con un producto mucho más refinado y depurado. En esta instancia los estudiantes además de contar y mostrar su trabajo vuelven a intercambiar ideas con otros estudiantes y docentes de la carrera, buscando ideas para la realización del Proyecto II ya a esta altura presentado por los docentes. Se muestran algunas fotos de esta instancia del período 2017-2019 (presencial) y 2020-2021 (virtual) en la **Figura 3**.

En cuanto a las fortalezas de este tipo de metodologías, se puede señalar el poder observar con claridad el progreso de los y las estudiantes en cuanto a la adquisición de conocimiento y saberes disciplinares. Al ofrecer dentro del diseño de la materia múltiples instancias de encuentro, es posible contar con registros cotidianos del progreso, de la evolución, de las mejoras, los cambios realizados a partir de las correcciones dadas en las devoluciones, la indagación del grupo, etc. Asimismo, la implementación de encuestas al finalizar cada proyecto, aporta datos en cuanto a la vivencia del estudiantado, las dificultades transitadas, los caminos de resolución y las preferencias.

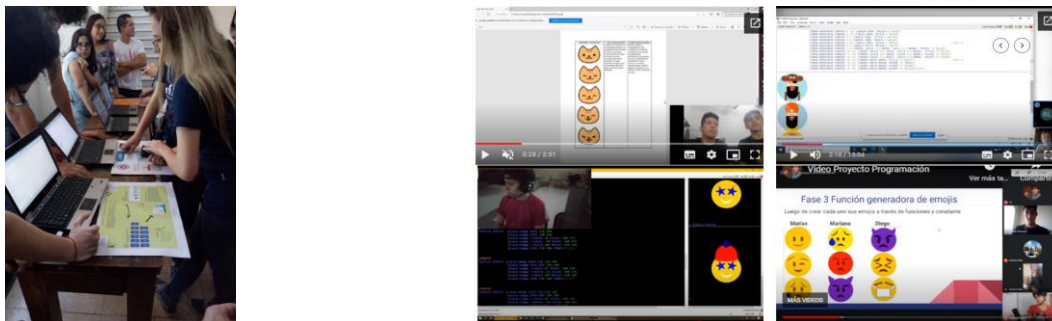
Por otro lado, y a raíz de la situación pandémica, se ha podido analizar cómo la estrategia del ABP resulta fácilmente adaptable a la virtualidad. En nuestro caso, esa adaptación ha implicado un rediseño particularmente de la metodología de evaluación y la exposición final, empleando una Vidriera Virtual de Exposición de Proyectos<sup>2</sup>, que nos permitió llevar adelante un trabajo de evaluación asíncrono entre pares y entre los docentes. Tomando como premisa que se cumplieran con los objetivos de evaluación pero que también fuera sencillo y rápido de desarrollar, se eligió

---

<sup>1</sup>Ver Coffee-Racket-Break // enlace oculto para cumplir con el anonimato

<sup>2</sup>Ver Vidriera de Exposición de Proyectos: <https://sites.google.com/view/vidriera-proyectos-fceia/>

utilizar la plataforma de Google Sites. Se optó por un diseño simple, con una página principal donde se presenta el aula del redictado de Programación, las docentes a cargo y muy brevemente el grupo de investigación "Computación con Ingenio" que las autoras han creado y constituido para pensar, discutir, evaluar, mejorar y consolidar los distintos proyectos de enseñanza de la programación y el Pensamiento Computacional desde el nivel universitario. Allí, como se observa en la Figura 4, cada grupo pudo contar con un espacio para sus proyectos, ofreciendo también un espacio para que el resto de estudiantes aporte comentarios.



**Figura 3.** Registros de Fotos de Exposiciones Presenciales y Virtuales de los Proyectos Desarrollados en el Redictado, bajo la metodología ABP

Otro aspecto interesante a destacar como fortaleza de la experiencia, es el hecho de que esta metodología permite evitar el plagio, que generalmente dentro de la enseñanza de la programación, suele ser un aspecto presente (Araujo y Kyrilov, 2020; Fraser, 2014). Por un lado, el hecho de que el trabajo a realizar tenga un aspecto creativo lo hace único y difícilmente encontremos en la web una solución directa a algo que se pensó entre un grupo de personas bajo las singularidades de cada uno de sus integrantes. A su vez, los problemas son pensados por las docentes variando las temáticas año tras año, lo cual hace que las referencias a los años anteriores no puedan hallarse fácilmente, no hay repositorios de trabajos de cursadas previas. Por otro lado, el trabajo grupal incentiva a la consulta entre pares como medio de alcanzar conocimiento, de lograr aprender a partir del trabajo conjunto. Esto previene la búsqueda de respuestas en otros, ya que los otros forman parte del grupo, y si bien son pares y están en el mismo nivel académico, el trabajo de una forma no aislada tracciona hacia adelante, hacia la resolución del problema.

Finalmente, la supervisión guiada semanal, bloquea también búsquedas que los estudiantes realizan en terceros para consultar, ya que el control semanal es ese espacio donde preguntar aquello que no se entiende o no se puede resolver por completo. Al encontrarnos para hablar sobre lo realizado y mostrar los avances, aquello que se detecta fuera de lugar, o un tipo de programación no acorde al nivel junior o inicial en el área, resalta rápidamente al ojo del docente y puede dialogarse con los estudiantes para indagar sobre su origen y el porqué de la búsqueda en ese tipo de respuesta y no otra acorde a lo pedido.

Las debilidades que se pueden señalar en la implantación de la técnica se encuentran vinculadas a la necesidad de contar con alguna herramienta específica para realizar los seguimientos de los grupos de trabajo y documentar el progreso cuando la cantidad de estudiantes supera los cincuenta. A su vez, es importante formar otros docentes y auxiliares para dominar la forma de trabajo de manera tal que puedan colaborar para realizar las tutorías de los proyectos, manejar los

grupos de trabajo frente a las dificultades que puedan surgir, y saber evaluar formativamente a los grupos en cada instancia de control. También resulta complejo que la innovación sea bienvenida dentro de las carreras o a los ojos de las autoridades, y se cuente con ello el apoyo genuino para continuar produciendo cambios en esa búsqueda, que muchos docentes tienen, de adaptarse a las nuevas necesidades de cada disciplinas, entonces muchas veces se prefiere continuar por un recorrido de aprendizaje y enseñanza conocido, aunque éste no logre cumplir con todas las expectativas que la disciplina hoy demanda a los espacio de formación superior.



**Figura 4.** La Vidriera de Proyectos, un Espacio de Socialización y Evaluación de los Proyectos Realizados en el Redictado bajo la Metodología ABP

## Conclusiones

Considerando lo expuesto en el presente trabajo, se observa que la implementación de la metodología del ABP en segundos cursados es una estrategia que habilita la construcción de aprendizajes significativos por parte de los y las estudiantes. El ABP permite superar las dificultades para acceder al contenido que presentan los alumnos recursantes, ayudándolos a aprender mediante el intercambio con sus compañeros y la colaboración con sus compañeros/as, favoreciendo el desarrollo de su autonomía y su creatividad. A medida que avanza el cursado, es posible identificar un incremento de la confianza para trabajar con problemas cada vez más complejos, desarrollando habilidades comunicativas, persistencia, tolerancia a la frustración y la ambigüedad, la habilidad para lidiar con los problemas abiertos y cerrados, etc.

Los proyectos están sustentados en las premisas del pensamiento computacional de (Wing, 2006) y las estrategias de resolución de problemas de (Polya, 1973) adaptadas por (Thompson, 1997) para el desarrollo de programas, estableciendo pilares disciplinares sólidos para el futuro en la carrera académica y profesional de los estudiantes. En este sentido, los cambios introducidos en las actividades didácticas de la cursada han permitido andamiar los procesos de aprendizaje de los contenidos específicos de programación trabajados en la cátedra, lo cual a futuro generarán confianza y robustez en los estudiantes para afrontar nuevos desafíos disciplinares.



A partir de las experiencias realizadas, los aprendizajes alcanzados y las fortalezas detectadas, consideramos fundamental continuar con la implementación de esta estrategia en el redictado, ofreciendo a los y las estudiantes propuestas didácticas que les permitan resignificar lo trabajado en sus primeras cursadas. En esta dirección y con la misma filosofía de trabajo, se modificó el cursado del redictado de Programación II para LCC, donde se enseña a programar usando el lenguaje Python y C para implementar en dicha cátedra también ABP sobre un trabajo integrador en el lenguaje Python. La experiencia ganada como docentes sobre el uso y los alcances de la estrategia didáctica y el dominio de la técnica del ABP ha permitido lograr rápidamente una nueva implantación de la estrategia didáctica con resultados igualmente de prometedores que en los otros cursos donde se trabajó la estrategia y que al finalizar dicha experiencia también documentaremos y analizaremos.

## Bibliografía

- Araujo, G. G., & Kyrilov, A. (2020). Plagiarism prevention through project based learning with GitLab. *Journal of Computing Sciences in Colleges*, 35(10), 53-58.
- Bloch, S. (2010). *Picturing Programs, An Introduction to Computer Programming*. College Publications.
- Colussi, N. & Viale, P. (2019). Actividades de Programación Grupales para Primer año de la Licenciatura en Ciencias de la Computación. Experiencias Didácticas en el Aula. En Pairoba, C., Cricco, J. y Rius, S. (comp). *Actas de XIII Jornadas de Ciencia y Técnica* (pp. 196). Universidad Nacional de Rosario.
- Colussi, N. & Viale, P. (2020) Proyecto de Investigación Biental: “Estrategias Didácticas para el Aprendizaje y la Enseñanza del Pensamiento Computacional en el Nivel Académico Universitario”. NRO: 80020190100255UR. SECyT-FCEIA. UNR.
- Domínguez, J. A., Carod, E. S., y Velilla, M. J. (2008). Comparativa entre el aprendizaje basado en proyectos y el aprendizaje basado en problemas. *II Jornadas de innovación docente, tecnologías de la información y de la comunicación e investigación educativa*. Zaragoza.
- Felleisen, M., Findler, R., Flatt, M. y Krishnamurthi, S. (2001). *How to Design Programs: An Introduction to Programming and Computing*. MIT Press: USA.
- Fraser, R. (2014). Collaboration, collusion and plagiarism in computer science coursework. En *Informatics in Education*, Volumen 13 (pp. 179-195). DOI: 10.15388/infedu.2014.01
- García Martín, J., Perez Martínez, Jorge. (2018). Aprendizaje basado en proyectos: método para el diseño de actividades. *Revista Tecnología, Ciencia y Educación*, 5, pp 37-63.
- Gómez, B. R. (2005). Aprendizaje basado en problemas (ABP): una innovación didáctica para la enseñanza universitaria. *Educación y educadores*, (8), 9-20.
- Heydrich, M., Rojas, M., & Hernández, A. (2010). Aprendizaje basado en proyectos: una experiencia de innovación docente. *Revista Universidad EAFIT*, 46(158), 11-21.
- Martinez Lopez, P. E. (2013). *Las Bases conceptuales de la Programación: una nueva forma de aprender a programar*. Universidad Nacional de Quilmes. <http://www.gobstones.org> > bibliografía > Libros
- Polya, G. (1945,1957,1973) *How To Solve It: A New Aspect of Mathematical Method*. Princeton: University Press.
- Queiruga-Dios, M. A.; Sáiz-Manzanares, M. C., y Montero-García, E. (2019). Problemas-Proyectos Adaptativos y Creativos en la enseñanza de las ciencias. Descripción de la metodología y apreciación de los estudiantes involucrados. *Research in Education and Learning Innovation Archives*, 23,1--23. 10.7203/realia.23.15567
- Sanchez, P. y Blanco, C. (2012). Implantación de una metodología de aprendizaje basada en proyectos para una asignatura de Ingeniería del Software. En *Actas XVIII JENUI 2012*, Ciudad Real, Universidad Nacional de Cantabria, España.
- Thompson S. (1997) Where do I begin? A problem solving approach in teaching functional programming. En Glaser, H., Hartel, P. y Kuchen, H. (eds.) *Programming Languages: Implementations, Logics, and Programs. PLILP 1997. Lecture Notes in Computer Science* (vol 1292). Springer, Berlin, Heidelberg.
- Wing, J. M. (2006). Computational thinking. *Commun. ACM*, 49(3), pp. 33-35.

# Aprender programación usando bloques y texto en forma simultánea

Un enfoque espiralado

Gonzalo Pablo Fernández<sup>1</sup>, Pablo E. “Fidel” Martínez López<sup>1</sup> y Cecilia Martínez<sup>2</sup>

<sup>1</sup>Universidad Nacional de Quilmes, Bernal, Argentina.

<sup>2</sup>Universidad Nacional de Córdoba, Córdoba, Argentina.

2022

**Resumen:** En este trabajo se presenta un enfoque particular para la enseñanza inicial de la programación usando tanto un entorno visual como un lenguaje textual. Aunque muchos trabajos anteriores proponen enfoques “híbridos” que combinan ambos métodos, la mayoría falla en mantener la cohesión entre los primeros y los segundos, resultando en dificultades en el aprendizaje. A diferencia de estos, el enfoque propuesto no distingue dos etapas aisladas sino que usa ambas herramientas constantemente y así evita el salto de dificultad que se genera al cambiar de paradigma.

**Palabras-clave:** Introducción a la programación, Programación por bloques, Enfoque espiralado, Gobstones.

## 1. Introducción

Hacia finales de los años '60 Seymour Papert crea LOGO, el primer lenguaje de programación diseñado específicamente para enseñar a programar. Durante muchos años fue el único lenguaje de programación con esa característica pero desde principios de milenio –propulsada por un interés global en la educación de contenidos relacionados a las ciencias de la computación y, en particular, a la programación– comienza una ola de desarrollo de lenguajes y entornos de programación educativos. Estos lenguajes y entornos buscan facilitar el aprendizaje de la programación simplificando determinados aspectos generalmente considerados complejos. De esta forma, la programación dejó de ser una disciplina reservada únicamente para la gente que trabaja en la industria informática y pasó a ser (o más bien, está en proceso de pasar a ser) parte de la educación básica general que toda persona debería recibir.

Entre las estrategias elegidas para facilitar la enseñanza de programación sin duda la más popular es la utilización de entornos visuales que permiten construir programas a través de manipulaciones icónicas en lugar de utilizando texto, como venía siendo hasta entonces. La mayoría de estos entornos visuales se basa en lo que ahora se conoce como *programación por bloques* que consiste en utilizar abstracciones visuales de segmentos de código (llamados *bloques*) que pueden combinarse y encastrarse sobre un lienzo para describir un comportamiento. Esto se traduce en un mecanismo de programación que limita las formas en que se pueden combinar las herramientas del lenguaje anulando así cualquier posibilidad de error sintáctico. Esta característica de los entornos de programación por bloques es lo que ha permitido que se popularicen tanto.

Desde que surgieron, los entornos visuales se han usado en una gran cantidad de contextos, destacando los contextos escolares en todos los niveles educativos. Su atractivo parece evidente: es más fácil

aprender cuando es más fácil la tarea a realizar. Al restringir la sintaxis y focalizar únicamente en la semántica, la carga de aprendizaje se reduce considerablemente. Sin embargo, la mayoría de los estudios realizados con estos entornos no parecen enfocarse en la mejora obtenida en el aprendizaje sino en la motivación que generan en el público (Malan y Leitner, 2007; Wagner y col., 2013). Si bien es cierto que la motivación puede indirectamente favorecer el aprendizaje, son pocos los estudios que efectivamente evalúan cómo los entornos visuales hacen que las personas aprendan de manera profunda, significativa, relacionando los conceptos y transfiriéndolos a otras experiencias.

De hecho, al leer los artículos que presentan estos entornos y sus principales características, se advierte mucho énfasis en la capacidad que tiene la herramienta para motivar las personas (Cooper y col., 2000; Maloney y col., 2010). Esto genera el interrogante de si es conveniente usar como instrumento de enseñanza herramientas que ponen tanta atención en el aspecto motivacional y no tanto en cumplir ciertos objetivos didácticos y pedagógicos. Muchos estudios sugieren que sí, comparando el aprendizaje utilizando entornos visuales con el aprendizaje a través de lenguajes textuales.

En este trabajo nos cuestionamos esa afirmación analizando la estructura de los enfoques didácticos utilizados en los estudios mencionados. En particular, observamos que los dictados de las distintas secuencias fallan en establecer claramente la relación entre lo que se aprende utilizando los entornos visuales y los conceptos de programación subyacentes. Nuestra hipótesis sugiere que la causa de esta falencia radica en la falta de un enfoque que integre correctamente ambas etapas del aprendizaje (aprender a usar el entorno por un lado y aprender los conceptos de programación por otro).

En consecuencia, presentamos un enfoque propio, poco explorado en la literatura, que venimos utilizando en cursos de programación durante los últimos 5 años, para aprovechar las ventajas de los entornos visuales en el aprendizaje, pero priorizando la calidad de los conocimientos adquiridos. Este enfoque está orientado principalmente a la educación universitaria pero es fácilmente adaptable a otros niveles educativos. El enfoque toma ideas ya exploradas para combinar el uso de entornos visuales con lenguajes textuales en un mismo curso pero se basa en el enfoque espiralado para combinarlos, resultando en una propuesta única que mantiene la cohesión entre ambos y minimiza el salto de dificultad que genera en muchos de los enfoques existentes el pasaje entre uno y otro.

Actualmente se están realizando estudios más profundos para evaluar los efectos del enfoque propuesto en el aprendizaje. En el presente trabajo solo se describe la motivación detrás del mismo y se realiza una comparación cualitativa con enfoques similares. Se espera en los próximos meses comenzar a realizar intervenciones para obtener resultados más concretos que permitan evaluar efectivamente las ventajas y desventajas del enfoque propuesto. En la sección 2 hacemos un relevamiento de trabajo previo en estudios sobre enseñanza de la programación utilizando entornos visuales para mejorar el proceso de aprendizaje. En la sección 3 describimos nuestra propuesta y en la sección 4 la comparamos con las propuestas existentes, analizando qué mejoras aporta.

## 2. Aprender con bloques: ventajas y problemáticas

Enseñar a programar es difícil. Aprender a programar requiere dominar distintas habilidades. El aprendizaje suele costar porque se suelen enseñar todas las habilidades juntas y es demasiado conocimiento como para asimilarlo todo a la vez (Xie y col., 2019).

Una de las habilidades que se deben aprender al programar es la escritura de código, para lo cual se debe aprender la sintaxis específica de algún lenguaje de programación. Al ser los lenguajes de programación tan acotados y estructurados, esta habilidad es difícil de adquirir para alguien que no tiene ninguna experiencia con lenguajes formales. Es muy común en principiantes entender cómo resolver un problema pero no saber cómo escribirlo en un lenguaje de programación.

Es esta problemática la que motiva el uso de entornos visuales basados en bloques. Al abstraer la estructura sintáctica del lenguaje, evita que personas sin experiencia comenten errores sintácticos al crear un programa. Las formas de combinar los bloques restringen su uso para que cualquier construcción sea sintácticamente válida. Utilizando bloques se puede ignorar la sintaxis y aprender únicamente las *construcciones programáticas* del lenguaje (es decir, las herramientas básicas del lenguaje con las cuales se construye un programa, tales como secuencias, alternativas y repeticiones, tanto en la categoría de los comandos como en la de las expresiones). Esta característica de los entornos de programación por bloques pareciera ser suficiente para justificar que son ideales para enseñar a programar a principiantes

pero, como ya se mencionó, es parte de aprender a programar aprender la sintaxis de un lenguaje de programación formal. Es válido preguntarse entonces si las personas que aprenden a programar usando entornos de programación por bloques están aprendiendo a programar en forma completa.

Ante este interrogante, muchos estudios proponen el uso de bloques simplemente como un medio temporal para facilitar el desarrollo de otras habilidades relacionadas al aprendizaje de la programación y retrasar el aprendizaje de la sintaxis para una etapa posterior (Armoni y col., 2015; Lewis, 2010; Malan y Leitner, 2007). Esta propuesta es interesante por el hecho de que permite entrenar distintas habilidades en distintas etapas del aprendizaje, reduciendo así el impacto ya mencionado percibido por quienes aprenden a programar por primera vez y deben enfrentarse a tantos aprendizajes simultáneos. Así surgen los llamados *enfoques híbridos* que utilizan entornos de programación por bloques para enseñar contenidos semánticos y luego pasan a un lenguaje de programación textual para enseñar sintaxis.

A pesar de lo intuitivo que suena esta estrategia, no todos los estudios que utilizan un enfoque híbrido lograron llegar a resultados concluyentes. En muchos casos se observan leves mejoras en el aprendizaje (tanto cuantitativas, en los resultados de las calificaciones en los exámenes como cualitativas, en las percepciones sobre el entendimiento de los conceptos) pero la mayoría detecta una dificultad para asociar los contenidos aprendidos durante la primera etapa a los problemas trabajados durante la segunda, evidenciando una falencia en el proceso de transferencia de conocimiento entre ambas etapas. Además, el uso de bloques como primer acercamiento a la programación ha demostrado ser menos estable en el sentido de que puede llevar a malentender ciertos conceptos o a adoptar malas prácticas de programación que luego son difíciles de desarraigar, como se verá en la sección 2.4.

## 2.1. Trabajo previo con el enfoque híbrido

En esta sección relevamos algunos de los estudios que utilizan el enfoque híbrido para mejorar el aprendizaje introductorio de programación. La mayoría de estos compara cómo es el aprendizaje de un lenguaje de programación textual para personas con y sin experiencia previa en un entorno de programación por bloques.

Malan y Leitner, 2007 usaron **Scratch** como puerta de entrada a la programación previo a enseñar **Java**. Su principal justificación es enseñar primero semántica (las construcciones programáticas) y resolución de problemas abstrayendo la sintaxis de un lenguaje textual para que luego, al aprender **Java**, solo haya que aprender la sintaxis. Dictaron un curso de 8 semanas con 25 estudiantes. Cada semana se dictaron dos clases de 2.5 horas cada una. La primera semana la dedicaron a **Scratch**, enseñando construcciones programáticas y lógica de resolución de problemas. Luego enseñaron **Java**. El foco no estuvo en mejorar el rendimiento en los exámenes sino en mejorar la experiencia del primer acercamiento a la programación. Por lo tanto, los resultados del estudio no apuntan a demostrar mejoras en el aprendizaje sino a evaluar los efectos positivos o negativos que el entorno visual pueda tener en la percepción posterior al trabajar con el lenguaje textual. Al final del curso hicieron una encuesta en la que el 76% opinó que **Scratch** fue una experiencia positiva y solo el 8% que fue negativa. El restante 16% opinó que no afectó en nada. Curiosamente, esas personas ya tenían experiencia previa en programación.

Armoni y col., 2015 armaron un curso introductorio de computación usando **Java** y **C#**. Participaron 120 estudiantes de entre 15 y 16 años, combinando estudiantes que habían tenido experiencia previa con **Scratch** y estudiantes que no habían tenido ninguna experiencia previa en programación. A nivel calificaciones no encontraron grandes diferencias entre ambos grupos pero sí percibieron diferencias cualitativas entre las que se destacan mayor facilidad para asimilar conceptos y menos tiempo necesario para entenderlos por parte de quienes ya habían trabajado con **Scratch**. También detectaron mayor confianza y motivación en ese grupo.

Lewis, 2010 comparó dos posibles caminos en un curso de 12 clases de 3 horas cada una con estudiantes de entre 10 y 12 años. Los contenidos dictados en cada clase fueron los mismos para ambos grupos pero un grupo usó **Scratch** durante las primeras 6 clases y **LOGO** durante las 6 restantes y el otro grupo al revés. Aunque percibió que el grupo que empezó con **Scratch** tuvo menos dificultades en el aprendizaje, no hubo diferencias significativas en el rendimiento en los exámenes.



## 2.2. Ventajas de aprender con bloques

La principal ventaja de utilizar un entorno de programación por bloques como primer contacto con la programación, respecto a utilizar un lenguaje de programación textual, probablemente sea la abstracción de la estructura sintáctica de las construcciones programáticas. Para quien nunca ha utilizado un lenguaje formal, puede resultar anti-intuitiva la necesidad de describir un comportamiento teniendo que respetar tantas reglas. Al utilizar bloques en lugar de texto, esta compleja carga desaparece, ya que las formas posibles de combinar bloques para crear construcciones más complejas están restringidas únicamente a combinaciones sintácticamente válidas, mientras que en un lenguaje textual se permite escribir cualquier combinación de caracteres. Esta ventaja fue destacada por Malan y Leitner, 2007 pero se menciona en muchos artículos que presentan entornos visuales (Maloney y col., 2010).

La representación visual a través de imágenes en lugar de texto también simplifica la composición. Otra ventaja de la programación por bloques, destacada por Weintrop, 2015, es el mecanismo más intuitivo de construcción de programas (comúnmente conocido como *composición drag-and-drop*) a través del cual se selecciona la construcción del lenguaje que se quiere incorporar y se la arrastra al espacio de composición donde se encuentra el programa definido hasta el momento (el equivalente al campo de texto en el que se escribe un programa cuando se trabaja con lenguajes textuales). Para poder permitir este comportamiento, el área de trabajo viene acompañada de una paleta (comúnmente llamada *toolbox*), desde donde se pueden ver las construcciones disponibles que se pueden incorporar al programa en desarrollo.

En relación a esto último, Weintrop y Wilensky, 2015 también destacan como ventaja la existencia de este *toolbox*, que permite visualizar todas las construcciones del lenguaje que están disponibles. Esto evita la necesidad de tener que memorizar cuáles son pero también la necesidad de tener que memorizar cómo se escriben (es decir, las palabras clave del lenguaje que se asocian a cada una de ellas en un lenguaje textual).

Otra ventaja destacada por Weintrop y Wilensky, 2015 es la legibilidad, ya que los textos sobre los bloques suelen componer oraciones mucho más cercanas al lenguaje natural que las resultantes al componer las palabras en el código de un lenguaje textual. Esta característica sin embargo depende del lenguaje textual con el que se compare ya que la sintaxis puede ser muy variada. Sí puede apreciarse esta ventaja al trabajar con estudiantes que no usan el inglés como idioma nativo, teniendo en cuenta que la gran mayoría de los lenguajes textuales pueden ser escritos exclusivamente en inglés, mientras que en los entornos de bloques suele ser muy sencillo admitir múltiples traducciones, acercando la programación de forma familiar a personas de todo el mundo.

## 2.3. Transferencia

Las ventajas descritas en la sección anterior parecerían indicar que aprender a programar utilizando entornos de programación por bloques es bastante conveniente. Sin embargo, al haber sido diseñados como herramientas educativas, estos entornos suelen ser bastante restrictivos, por lo que no suelen ser usados en el ámbito profesional. Es esperable entonces que la enseñanza de la programación no se limite únicamente a entender los mecanismos semánticos de construcción de programas sino que también enseñe a lidiar con la sintaxis específica de un lenguaje industrial (o al menos, uno basado en texto). En base a esto es que surgen tantos enfoques que plantean enseñar ciertos conceptos con entornos visuales para luego pasar a utilizar lenguajes textuales, como los descritos en la sección 2.1. Todos estos se basan en la idea (no necesariamente cierta) de que todo el conocimiento aprendido utilizando el entorno visual se transfiere trivialmente al comenzar a utilizar el lenguaje textual.

Aunque no haya resultados concluyentes, en la mayoría de los estudios antes mencionados aparecen indicios de que los conceptos adquiridos durante la etapa de programación por bloques pueden ser transferidos, al menos en parte, a la etapa de programación textual (Armoni y col., 2015; Dann y col., 2012; Malan y Leitner, 2007; Wagner y col., 2013). Malan y Leitner, 2007 por ejemplo recogen algunos testimonios que muestran que un concepto a ser utilizado en el lenguaje textual puede ser asociado correctamente al bloque correspondiente: “aunque no sabía aún cómo crear un ciclo, sabía cuándo un ciclo era necesario porque había usado ciclos en mis programas de Scratch”<sup>1</sup>.

---

<sup>1</sup>Traducción propia. Cita original: *Though I did not yet know how to create a for loop, I knew when a for loop was necessary because I had used loops in my Scratch programs.*

Por otro lado, algunos de los estudios también advierten mayor facilidad para asimilar nuevos conceptos en quienes tuvieron experiencia previa en programación usando un entorno visual respecto a quienes no tuvieron ninguna experiencia en absoluto (Armoni y col., 2015; Tabet y col., 2016).

## 2.4. Desventajas de aprender con bloques

Los beneficios aportados por la enseñanza de la programación a través de los entornos visuales que detectaron los trabajos previos impulsan a considerarlos seriamente como herramientas de aprendizaje. Sin embargo, a pesar de todas las ventajas mencionadas en la sección 2.2, existen también problemáticas derivadas de estos enfoques que pocos estudios pudieron atender correctamente.

Meerbaum Salant y col., 2011 por ejemplo detectaron dos problemas muy grandes en la forma de pensar de quienes aprenden a programar usando **Scratch**. El primero es lo que denominan “pensamiento operacional” o “bottom-up extremo” y lo describen a través de la metodología que utilizaban quienes estaban aprendiendo. Esta metodología consistía en tomar varios bloques del *toolbox* que creían que podían llegar a necesitar y arrastrarlos al espacio de composición para luego empezar a pensar cómo combinarlos para lograr el comportamiento deseado. Este mal hábito (como lo llaman) contrasta significativamente con el enfoque comúnmente utilizado al enseñar a programar, según el cual, debe pensarse primero qué hacer, idear una estrategia de solución y por último estructurar el código para implementar la estrategia planificada (lo que se conoce normalmente como “top-down”).

Al segundo mal hábito lo denominan “Programación extremadamente granular” (*EFGP* por sus siglas en inglés, *Extremely Fine-Grained Programming*) y consiste en armar porciones de código que se encarguen de tareas extremadamente particulares. A este comportamiento lo ejemplifican con una solución propuesta para un recorrido en el que desde 3 porciones de código aisladas se controlan distintos aspectos de un recorrido basado en repetición condicional (el cuerpo por un lado, la condición de corte por otro y la inicialización en un tercero).

Al comparar un curso de estudiantes que aprendió a programar con **Scratch** y otro que aprendió a programar con **LOGO**, Lewis, 2010 notó que el primer grupo se tenía menos confianza en cuanto a sus habilidades programáticas. La hipótesis que propone para explicar tal situación es que los entornos visuales generan la sensación de que quienes lo utilizan no están programando realmente sino que solo están jugando. Si bien el aspecto lúdico es importante para mantener la motivación en el aprendizaje, en este caso parece estar teniendo una contraparte negativa que no puede pasarse por alto.

Se ha destacado como ventaja el hecho de que los bloques, como representación visual de las construcciones programáticas, se encastren como piezas de rompecabezas o ladrillos encastrables. Se supone que esta analogía con elementos físicos tangibles ayuda a manipularlos intuitivamente. Sin embargo, Weintrop y Wilensky, 2015 descubrieron que la analogía con el rompecabezas también puede generar la falsa idea de que existe una única forma correcta de resolver un problema dado utilizando bloques, ya que eso es lo que sucede en el caso de los rompecabezas. Esto surge a partir de un testimonio particular que asegura que “cada cosa tiene su lugar” (*everything has its place*) como si la solución ya existiera pero estuviera oculta y su tarea no fuera inventarla sino descubrirla encontrando la ubicación que le corresponde a cada pieza del rompecabezas.

Otra desventaja mencionada por Weintrop y Wilensky, 2015 es el hecho de que muchas actividades pensadas para programación por bloques consisten únicamente en desafíos de completar lo que falta, provocando el desinterés, el aburrimiento o la sensación de que no se está aprendiendo realmente. Si bien no es una desventaja del paradigma de bloques en sí, sino de las actividades planificadas, es cierto que sus características motivan la realización de este tipo de actividades.

## 2.5. El “gap” entre bloques y texto

Todas las propuestas descriptas en la sección 2.1 distinguen claramente dos etapas de aprendizaje. Comienzan con una primera etapa en la que trabajan exclusivamente en el entorno visual y en determinado momento ‘hacen el salto’ al lenguaje textual. No todas coinciden en cuáles conceptos deben enseñarse en cada etapa. Algunas elijen ciertos conceptos más simples para enseñar durante la etapa de bloques y dejan los más complejos para la etapa textual. Ese tipo de enfoque usa los entornos visuales como un lenguaje más, introductorio, para no comenzar a escribir texto directamente. Otras propuestas enseñan lo mismo en cada etapa pero destacando la forma de describir el comportamiento en cada una (a través

de bloques o a través de texto). Este segundo enfoque parece aprovechar mejor el uso de los bloques ya que los usa como herramienta para introducir un concepto más allá de su sintaxis particular y luego, al pasar a la etapa textual, solo debe enseñarse la sintaxis de cada una de las herramientas ya aprendidas en la primera etapa.

Podríamos decir que este enfoque distingue la enseñanza de la semántica por un lado y la de la sintaxis por otro. Como ya se mencionó, la abstracción de la sintaxis es una de las principales (y más aprovechadas) ventajas de los entornos de programación por bloques. Las propuestas que utilizan este segundo enfoque aprovechan esta ventaja para enseñar solo la parte semántica de cada una de las construcciones programáticas durante la primera etapa para luego enseñar únicamente la sintaxis del lenguaje textual en la segunda. El primer enfoque, en contraposición, parece usar los bloques únicamente para facilitar el inicio a principiantes pero una vez que pasa a la etapa textual no se detiene en la distinción entre sintaxis y semántica sino que parece asumir que el conocimiento adquirido durante la primera etapa permite el desarrollo y maduración suficientes para aprender ambos aspectos a la vez.

Ya sea que usen el primer enfoque o el segundo enfoque, todas las propuestas tienen que lidiar con un problema importante que consiste en determinar en qué momento del aprendizaje es conveniente realizar ese pasaje entre la primera etapa y la segunda, y más importante aún, cómo hacerlo. Varias de las propuestas antes analizadas advierten este problema y hablan de estrategias para “cerrar el gap” y “suavizar la transición”. Dann y col., 2012 en particular utilizan *Mediated Transfer* para mejorar la transferencia de conceptos entre lo aprendido en la etapa de bloques y lo aprendido en la etapa textual. Si bien los resultados muestran una considerable mejora al aplicar estas técnicas, sigue quedando trabajo por hacer.

### 3. Nuestra propuesta

La propuesta que se describe en esta sección surge de sintetizar el esquema de clases que se plantea en la materia *Introducción a la Programación* de la *Universidad Nacional de Quilmes* (UNQ) desde 2017. La materia sigue el enfoque **Gobstones** (Martínez López y col., 2017) desde 2009 y ha servido como guía para el desarrollo tanto de la didáctica de **Gobstones**, como del lenguaje en sí. A lo largo del curso se van presentando cada una de las construcciones programáticas en tres etapas. La primera etapa es la *indagación* que consiste en un primer acercamiento a la herramienta que se va a presentar a partir de actividades que generen la necesidad de la herramienta en cuestión. Estas actividades presentan un problema que no se puede resolver con las herramientas vistas hasta el momento o que sí se puede pero la forma de solucionarlo es bastante engorrosa. La segunda etapa es la explicación teórica respecto a la nueva herramienta en la que se le da un nombre, se ejemplifican escenarios de uso y se formaliza su estructura, entre otras cosas. La tercera etapa es la ejercitación práctica en la que se proponen distintos ejercicios que aprovechen las características de la nueva herramienta en base a lo aprendido en la etapa teórica y sea necesario además combinar la nueva herramienta con las herramientas anteriores.

#### 3.1. El enfoque Gobstones

Además de ser un lenguaje de programación, **Gobstones** es un enfoque didáctico cuyo principal objetivo es reforzar la abstracción y el pensamiento abstracto (Martínez López y col., 2017). Por ello, ofrece una visión denotacional de la programación, menos explorada que la visión operacional que ofrecen la mayoría de los enfoques tradicionales. Un ejemplo de esto es la forma de concebir a las variables como etiquetas que se le asignan a valores (similar a la definición matemática de variables) y no como celdas de memoria que es una definición atada a la implementación de bajo nivel. Además, siendo este un concepto tan abstracto, se retrasa su presentación hasta la etapa final del curso, a diferencia de los enfoques tradicionales que las introducen casi al principio.

Otro aspecto interesante es el foco en la legibilidad y la declaratividad a través de la división en subtarefas. En lugar de orientar el curso a buscar las soluciones más eficientes, se motiva a confeccionar soluciones bien organizadas (en cuanto a estructura sintáctica y semántica), modulares y que sean fáciles de entender y corregir (tanto por la persona que confeccionó la solución como por otras), dejando la eficiencia en un plano secundario. Para esto es crucial introducir la noción de procedimiento cuanto antes, y no dejarlo hacia el final del curso como suelen hacer los enfoques tradicionales. El uso de procedimientos

para lograr una buena división en subtareas es clave para desarrollar las habilidades de abstracción desde el principio.

Si bien el enfoque **Gobstones** es independiente del lenguaje de programación utilizado, el lenguaje **Gobstones** permite aprovechar al máximo las virtudes del enfoque. El modelo de computación del lenguaje **Gobstones** se basa en realizar operaciones sobre un tablero. Esto habilita comenzar a desarrollar las habilidades de programación a través de acciones concretas (comandos) que se traducen en alteraciones visibles sobre el estado del tablero. A medida que se avanza en el curso, las actividades se enfocan cada vez más en la descripción de operaciones sobre datos (expresiones) a través de funciones.

Otra de las características que hacen del lenguaje **Gobstones** ideal para llevar a cabo el enfoque **Gobstones** es la capacidad de generar vestimentas que permiten representar elementos adicionales sobre el tablero utilizando alguna codificación personalizada. Esto contribuye en gran parte a fomentar la capacidad de abstracción, obligando a tener que considerar la representación de elementos abstractos de un dominio a través de elementos concretos del universo de discurso de **Gobstones**.

### 3.2. Incorporando bloques

En 2015 surge una nueva versión del entorno de programación de **Gobstones** llamado **Gobstones Web**. Entre sus más destacadas características sobresale la posibilidad de programar utilizando bloques. La materia adopta rápidamente el uso de bloques para la etapa de indagación, permitiendo que el descubrimiento de la nueva herramienta no dependa de la asistencia docente ya que esta se traduce en la aparición de un nuevo bloque en el *toolbox* que cualquier estudiante puede encontrar por su cuenta en el momento en que le surge la necesidad de generar cierto comportamiento que no podía describir hasta entonces.

El esquema de clases de la materia pasa entonces a tener un formato similar al de las propuestas híbridas mencionadas en la sección 2.1, en el que un mismo curso combina un entorno de programación por bloques con un lenguaje de programación textual. Sin embargo, hay una diferencia clave entre este enfoque y todos los mencionados anteriormente. Los anteriores distinguían dos etapas en el curso, trabajando en la primera exclusivamente con bloques y en la segunda exclusivamente con texto, mientras que el enfoque aquí presentado utiliza ambos mecanismos de descripción de programas (bloques y texto) a la vez. La incorporación de cada nueva construcción programática durante el curso consiste entonces en una primera etapa con actividades dentro del entorno de bloques en las que se descubre la funcionalidad que provee la nueva herramienta (es decir, su semántica) y las formas en que esta puede combinarse con las demás herramientas conocidas, una segunda etapa de formalización y presentación de la sintaxis específica en el lenguaje textual (una vez comprendida la semántica de la misma) y una tercera etapa de ejercitación en el lenguaje textual. De esta forma, la introducción de una nueva herramienta se divide en la presentación de su estructura semántica primero y su estructura sintáctica luego.

## 4. Discusión

Todos los enfoques descritos en la sección 2.1 combinan un entorno de programación por bloques con un lenguaje de programación textual en un curso introductorio de programación. En todos los casos aparece el problema del “salto” entre el primero y el segundo. Algunos enfoques pudieron resolverlo mejor que otros, usando técnicas como *Mediated Transfer* pero ninguno logró cerrar el “gap” por completo. Proponemos que la raíz del problema es que se requiere un alto nivel de entrenamiento en abstracción para entender que ambos mecanismos de descripción de programas son equivalentes y que solo son formas distintas de expresar lo mismo.

La gran ventaja del enfoque utilizado en la materia *Introducción a la Programación* es que ese “salto” no se produce en algún momento del curso a partir del cual se modifica el mecanismo estándar que debe utilizarse para traducir a código las soluciones propuestas. Alternativamente, se puede pensar como que el “salto” se realiza constantemente, al introducir cada nueva herramienta del lenguaje.

La materia sigue el esquema del enfoque espiralado. En este esquema los conceptos son revisados reiteradas veces en un mismo curso, cambiando el enfoque o profundizando en detalles para asegurar una completa comprensión de los mismos. En nuestro caso particular, cada concepto se descubre a través de

su semántica, en el entorno de bloques y luego se vuelve a presentar desde un ángulo sintáctico en el lenguaje textual.

En las propuestas antes mencionadas no se aprovecha del todo esta característica de los bloques que permite enseñar una construcción programática desde dos ángulos distintos (semántica y sintaxis). Las que usan bloques solo para facilitar la introducción pero que no vuelven a revisar los conceptos enseñados durante la primera etapa están perdiendo de vista el aspecto sintáctico de todo lo que se haya enseñado durante esa primera etapa. Al pasar a la segunda etapa el primer concepto estudiado no solo tiene la carga de este nuevo concepto sino el peso de aprender la sintaxis de todo lo que hasta el momento solo se conocía a través de su semántica. Las que eligen en cambio volver a ver los conceptos de la primera etapa en la segunda tienen un salto mucho menor ya que los conceptos vuelven a verse uno por uno pero a través del enfoque sintáctico en lugar del semántico, como se hizo en la primera. Este segundo enfoque se parece mucho más a nuestra propuesta excepto por el orden. En este enfoque se aprende primero toda semántica del lenguaje (a través de cada una de las herramientas) y luego toda la sintaxis. En nuestra propuesta se aprende cada una de las herramientas en orden pero estudiando, para cada una, primero su aspecto semántico y luego su aspecto sintáctico.

Al estar constantemente cambiando entre el entorno de bloques y el lenguaje textual, nuestra propuesta no se enfrenta al problema de determinar el punto ideal en el aprendizaje en el cual hacer efectuar el “salto” entre paradigmas ni cómo debe ser atendida esta transición. Esto deviene en un aprendizaje mucho más fluido que permite ver a la semántica y la sintaxis como dos partes de lo mismo y no como características aisladas.

## 5. Conclusiones y trabajo futuro

Propusimos una novedosa estrategia para aprovechar el uso de los entornos de programación por bloques en el aprendizaje inicial de programación. Otros enfoques que también hacen uso de tales entornos no aprovechan al máximo su principal característica que es la abstracción de la sintaxis. Algunos de estos pierden de vista las posibilidades que esto ofrece y simplemente lo usan para que los primeros pasos sean más sencillos pero luego pasan a un lenguaje textual. Otros parecen comprender mejor esa ventaja y enseñan primero toda la semántica y luego toda la sintaxis. Aunque estos parecen estar mejor encaminados, al igual que los primeros tienen que lidiar con el problema del “salto” entre paradigmas.

Nuestra propuesta no se enfrenta a ese problema al avanzar con ambos paradigmas a la vez, enseñando cada una de las construcciones programáticas primero en bloques, para entender su semántica y luego en texto, para aprender su sintaxis. Creemos que este enfoque permite aprender completamente cada una de las construcciones (tanto su semántica como su sintaxis) antes de pasar a la siguiente y por eso debe ser considerada en lugar de las propuestas que dividen el curso en etapas aisladas.

Queda pendiente como trabajo futuro realizar las intervenciones pertinentes que permitan obtener resultados cuantitativos respecto a los efectos que esta propuesta tiene sobre el aprendizaje inicial de la programación y compararlo con los enfoques analizados previamente.

## Referencias

- Armoni, M., Meerbaum Salant, O. & Ben-Ari, M. (2015). From Scratch to “Real” Programming. *ACM Transactions on Computing Education*, 14(4), 1-15. <https://doi.org/https://doi.org/10.1145/2677087/>
- Cooper, S., Dann, W., Pausch, R., Moreno León, J., Casali, A. & Barone, D. (2000). Alice: a 3-d tool for introductory programming concepts. *Journal of Computing Sciences in Colleges*, 15(4), 107-116. <https://dl.acm.org/doi/pdf/10.5555/364133.364161>
- Dann, W., Cosgrove, D., Slater, D., Culyba, D. & Cooper, S. (2012). Mediated transfer: Alice 3 to Java. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education, SIGCSE '12 - ACM*, 141-146. <http://www.alice.org/wp-content/uploads/2017/04/MediatedTransfer.pdf>
- Lewis, C. M. (2010). How programming environment shapes perception, learning and goals: Logo vs. Scratch. *Proceedings of the 41st ACM Technical Symposium on Computer Science Education, SIGCSE '10 - ACM*, 346-350.

- Malan, D. J. & Leitner, H. H. (2007). Scratch for budding computer scientists. *Proceedings of the 38th ACM Technical Symposium on Computer Science Education, SIGCSE '07 - ACM*, 223-227. <https://cs.harvard.edu/malan/publications/fp079-malan.pdf>
- Maloney, J., Resnick, M., Rusk, N., Silverman, B. & Eastmond, E. (2010). The scratch programming language and environment. *ACM Transactions on Computing Education*, 10(4), 1-15. <https://doi.org/https://doi.org/10.1145/1868358.1868363/>
- Martínez López, P. E., Ciolek, D., Arévalo, G. & Pari, D. (2017). The Gobstones method for teaching computer programming. *XXV Simposio de Educación Superior en Computación (SIESC'17), dentro de la XLIII Conferencia Latinoamericana de Informática (CLEI'17)*, 1-9.
- Meerbaum Salant, O., Armoni, M. & Ben-Ari, M. (2011). Habits of programming in Scratch. *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education, ITiCSE '11 - ACM*, 168-172.
- Tabet, N., Gedawy, H., Alshikhabobakr, H. & Razak, S. (2016). From Alice to Python. Introducing Text-Based Programming in Middle Schools. *Proceedings of the annual joint conference on Innovation and technology in computer science education ITiCSE '16 - ACM*. <https://doi.org/https://doi.org/10.1145/2899415.2899462/>
- Wagner, A., Gray, J., Corley, J. & Wolber, D. (2013). Using app inventor in a k-12 summer camp. *Proceeding of the 44th ACM technical symposium on Computer science education, SIGCSE '13 - ACM*, 621-626. <https://doi.org/https://doi.org/10.1145/2445196.2445377/>
- Weintrop, D. (2015). Minding the gap between blocks-based and text-based programming: Evaluating introductory programming tools. *Proceedings of the 46th ACM Technical Symposium on Computer Science Education, SIGCSE '15, ACM*.
- Weintrop, D. & Wilensky, U. (2015). To Block or not to Block, That is the Question: Students' Perceptions of Blocks-based Programming. *Proceedings of the 14th international conference on interaction design and children - ACM*, 199-208.
- Xie, B., Loksa, D., Nelson, G. L., Davidson, M. J., Dongsheng, D., Kwik, H., Hui Tan, A., Hwa, L., Li, M. & Ko, A. J. (2019). A theory of instruction for introductory programming skills. *29(2)*, 205-253. <https://doi.org/https://doi.org/10.1080/08993408.2019.1565235/>

## Hay piratas en la sala de informática

IPET 424. Villa Dolores. Cordoba

Prof Graciela Ivana Guzmán

guzmanivana@hotmail.com

### **Resumen.**

La enseñanza de la programación. Esta experiencia plantea desde una secuencia didáctica aprender a programar. Partiendo de un dibujo en papel, con figuras geométricas, retomando contenidos previos de matemática y geometría. Para luego, digitalizarlo mediante código de programación processing, aplicando también lo aprendido en la materia de Inglés y descubriendo la lógica que interviene en los dibujos.

La experiencia de aprendizaje se hace aún más fuerte e interesante mediante un juego de búsqueda del tesoro, cuyas pistas se encuentran en códigos QR. Y para poder avanzar solo deben reconstruir programas y resolver los acertijos.

**Palabras clave:** juegos processing programación secuencia didáctica, escuela secundaria

Introducción

### **Marco conceptual en que se basa la experiencia docente**

“Un juego lógico, debe reunir criterios que propicien el desarrollo del pensamiento computacional, estar formulado como problema o establecer un desafío en el que se necesiten realizar abstracciones y generalizaciones, procesar sistemáticamente la información y representarla simbólicamente, utilizar algoritmos, descomponer el problema en subproblemas, poner en acción un pensamiento interactivo, recursivo y paralelo, evidenciar procesos de lógica condicional, requisitos de eficiencia y performance, depuración y corrección de errores.

Estos criterios se establecen desde la perspectiva que promueve el desarrollo del pensamiento lógico matemático como fundamental para incorporar conceptos abstractos, razonamientos y comprensión de relaciones. Todas estas habilidades van mucho más allá de las matemáticas entendidas como tales, los beneficios de este tipo de pensamiento contribuyen a un desarrollo en general de la persona.

Por otro lado, la alfabetización digital también tiene que integrar nociones sobre los lenguajes de las computadoras, sobre cuya base están construidos los contenidos fundamentales de nuestra sociedad, particularmente aquellos ligados a los consumos culturales de niños y jóvenes. En este marco, tanto la programación y la robótica, como el pensamiento computacional resultan relevantes para el aprendizaje: al comprender su semántica y su lógica en la resolución de problemas, los estudiantes se preparan para entender y cambiar el mundo.

Y finalmente ¿Qué es Processing? Es un software de código abierto, y cualquier persona puede contribuir en su mejora. Desprovisto de interfaces innecesarias, se vale de un lenguaje de programación (de igual nombre) basado en Java para realizar composiciones de gran interés. Como software, consiste básicamente en un simple editor de texto para escribir código, un área de mensaje, una consola de texto, un sistema de pestañas para manejar archivos, una barra de

herramientas con botones de accionar común y una barra de menú. Cuando el programa se ejecuta, se abre una ventana de representación.

### **Descripción de la Experiencia**

Esta propuesta es una secuencia didáctica que se llevó adelante en 6to año, del IPET 424 de Villa Dolores. Los estudiantes de sexto año, son un grupo de 32 estudiantes, en el momento de la implementación, provenientes de la localidad de Villa Dolores (Cba) y algunos parajes cercanos. Tienen entre 17 y 18 años

Dentro del espacio curricular de Laboratorio de Programación, materia con 6 horas semanales, que en su primera unidad, incluye el desarrollo de imágenes vectoriales. Al ser una experiencia de programación con código escrito estos eran algunos objetivos planteados:

- Implementar estrategias propias y grupales, en el desarrollo de proyectos audiovisuales.
- Utilizar conocimientos adquiridos, acorde a la situación o desafío por resolver.
- Iniciarse en el pensamiento computacional a través de Processing, incorporando conceptos de programación

La secuencia comienza con un dibujo en un papel utilizando diferentes figuras geométricas. Retomando lo ya aprendido en geometría, sobre las propiedades de estas figuras, y sus nombres en inglés, comenzamos en processing, las primeras funciones.

Este es un primer contacto con la programación escribiendo en código, sin usar un lenguaje en bloques, con todo lo que implica, errores de tipeo, de sintaxis, de lógica.

Una serie de desafíos con dibujos hacían que el uso del lenguaje se tornara más complejo profundizando en espiral, y así se vislumbraron los primeros programas. También sucedía que cuando los estudiantes querían introducir más detalles como colores o bordes, iban intentando / descubriendo funciones o parámetros nuevos mediante prueba error. Así por ejemplo dibujaban en un principio una cancha de football o la bandera olímpica, llegando a dibujar un monstruo animado o un mandala que cambiaba de colores.

Pasadas unas clases más, salimos a programar al patio, a jugar a una búsqueda del tesoro. Todos éramos piratas de processing. Las pistas, estaban escondidas en el patio de la escuela. En códigos Qr. Que al leer el QR se recuperaba un escrito, un programa escrito en processing con muchísimos errores, que debían corregir y al ejecutar deducir un acertijo que los llevaba a la próxima pista y finalmente al tesoro. Los chicos jugaron en equipos, con una compu y un celular por equipo durante las dos primeras horas de clase.

Con esta experiencia, todos los estudiantes durante una hora programaron y aplicaron todo lo visto hasta ese momento, sin presiones, solo la intensidad del desafío. Fue divertido y se perdió la noción del tiempo durante el juego.

Como cierre en el patio, compartiendo una merienda, hubo un pequeño debate sobre la experiencia, lo que les gustó y lo que les resultó más difícil, cosas que descubrieron y como se sintieron.

En las clases siguientes, se hicieron otros ejercicios y desafíos en processing con estructuras más complejas, uso de variables, variables de sistema, ciclos repetitivos y por supuesto logrando dibujos y animaciones.

En cuanto a la evaluación de la actividad se realizó un seguimiento del proceso, con registro de observaciones para la evaluación formativa individual.

.

### **Resultados**



Los estudiantes, luego de pasar por esta experiencia, pueden codificar en processing, identifican variables de sistema, interpretan los mensajes de error y pueden intervenir para solucionarlos. Agudizaron la búsqueda en internet de soluciones o alternativas a los desafíos dados, buscando de manera específica y aplicando lenguaje técnico para agudizar la búsqueda.

En palabras de Thiago: “Entendí que hay que exactamente cada palabra para que funcione el programa” o como dijo, Manuel: “al fin entendí todo” o como dijo Lourdes: “nunca me gusto programar, y ahora sí”

También, es valioso reconocer que a medida que avanzaba el desarrollo de estas actividades, cada estudiante iba incorporando una actitud participativa, con la docente y con sus pares, expectante ante la propuesta, con escucha atenta y emitiendo opiniones, escuchando las de sus compañeros de manera positiva y con buena predisposición. Que mejoró la convivencia dentro del aula, al punto que luego se organizaron para realizar ventas de cosas dulces en la plaza.

## Conclusiones

Las políticas educativas intentan introducir contenidos de la enseñanza de la programación y la robótica en todos los niveles educativos. En las escuelas del interior con los pocos recursos que contamos, se agudiza el ingenio a la hora de pensar estrategias y métodos para enseñar.

Con juegos en el aula, compartiendo y escuchando a los jóvenes, se agrega a la enseñanza de la programación, el desarrollo del pensamiento computacional, el trabajo colaborativo, la enseñanza entre pares y el aprendizaje por descubrimiento. Nos ayuda a entender lo que pasa en el aula para mejorar nuestra practica y acercarnos a nuestros estudiantes.

También, como docente, quiero mirar con esperanza a los jóvenes, construir con ellos un nuevo conocimiento que los ayude a comprender el mundo que los rodea, salir de ser consumidores de tecnologías para empoderarse como constructores o creadores de tecnología. Y que los hará, seguramente mejores personas, más responsables y futuros ciudadanos críticos y útiles en su comunidad.

## Referencias

Echeveste, M. E. (2017). Situaciones escolares de jóvenes que aprenden programación: una posición activa del sujeto del aprendizaje. En XII Congreso de Tecnología en Educación y Educación en Tecnología. La Matanza: TE&ET. Recuperado de [http://sedici.unlp.edu.ar/bitstream/handle/10915/63396/Documento\\_completo.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/63396/Documento_completo.pdf?sequence=1).

Ministerio de Educación de la Nación Orientaciones pedagógicas de Educación Digital. - 1a ed. - Ciudad Autónoma de Buenos Aires: Ministerio de Educación de la Nación, 2017. Libro digital, PDF Archivo Digital: descarga y online ISBN 978-950-00-1199-0 1. Aporte Educativo. 2. Medios Digitales. I. Título CDD 004.071

Ignacio Buioli y Jaime Pérez Marin. Manual de Processing. Actualizado a la ed 2016 vr 03. Recuperado de

[http://laurence.com.ar/artes/comun/Processing\\_un\\_lenguaje\\_al%20alcance\\_de\\_todos.pdf](http://laurence.com.ar/artes/comun/Processing_un_lenguaje_al%20alcance_de_todos.pdf)

## **Keylogging y screencapture como herramientas de laboratorio para estudios de casos en ciencias de la computación.**

**Autor:** Lic. Valentín Basel, Universidad Nacional de Córdoba, valentinbasel@gmail.com

**Palabras claves:** domótica, bibliotecas populares, keylogging, screencapture, investigación cualitativa

### **Resumen**

En la observación tradicional de un curso, el investigador se “sienta en un costado del aula” con su cuaderno de notas y tal vez una cámara de filmación, para así observar las dinámicas de interacción entre docentes-estudiantes y llevar un registro de la actividad. Sin embargo, este ejercicio se encuentra con varios inconvenientes a la hora de dar cuenta de las dinámicas inherentes a curso de programación. Por un lado, si un hipotético investigador solo se dedica a filmar a un costado del aula, lo más probable es que solo vea un montón de estudiantes sentados en silencio al frente de sus computadoras; y si decide registrar más en detalle, deba acercarse individualmente a cada estudiante para poder ver que está pasando en esa interacción estudiante-computadora, dejando de lado a otros estudiantes o la interacción global del aula. La dinámica de trabajo de un curso de programación implica una fuerte interacción de los estudiantes con las computadoras, la actividad está pasando precisamente en el uso de software específico, como búsquedas en internet y participación en chats con otras personas que pueden no estar presencialmente en el aula.

El Keylogging y screencapture utilizado como instrumento de investigación, no solamente permite tener el “producto final” del desarrollo de software, sino también los pasos intermedios en su construcción. Además, da cuenta de las prácticas de escritura llevadas a cabo por los participantes del curso, siendo una excelente opción para documentar los procesos de desarrollo y corrección de software.

### **Introducción**

Cuando un estudiante trabaja sobre un proyecto de desarrollo de software, sobre todo usando las metodologías de enseñanza basadas en proyectos (Botella Nicolás & Ramos Ramos, 2019; Rodríguez-Sandoval et al., 2010), generalmente no conserva un registro sistemático y metódico que de cuenta del trabajo que fue gestando a lo largo de su proyecto de programación. Los docentes e investigadores nada más pueden ver el resultado final de la práctica llevada a cabo por los estudiantes, perdiendo todas las iteraciones intermedias donde los alumnos prueban y corrigen su código fuente, o las discusiones concernientes al desarrollo de proyecto. En ese sentido, existen soluciones de software para enseñanza que pueden conservar las distintas instancias de ejecución llevadas a cabo por los estudiantes. Esta práctica permite conservar todas las distintas versiones de un mismo proyecto o ejercicio, por ejemplo las plataformas mumuki o Gobstones permiten tener

versiones de control cada vez que un estudiante ejecuta su código fuente. Esta forma de trabajo nos ayuda a tener una gran capacidad de registro sobre el trabajo de los estudiantes, viendo como evoluciona su código fuente, proporcionando la capacidad de usar técnicas de analíticas de aprendizaje (Zapata-Ros, 2015) para hacer trabajos estadísticos sobre las prácticas de los estudiantes.

No obstante, ciertos aspectos del desarrollo cognitivo de los estudiantes quedan en segundo plano con respecto a un análisis cuantitativo de su interacción con estas plataformas de enseñanza. Cuestiones específicas como la capacidad de autocorregir sus errores de sintaxis (errores como presionar mal una tecla para escribir una instrucción o escribir mal un nombre y corregirlo antes de ejecutar el intérprete PYTHON) así como el tiempo que los estudiantes permanecen dentro de la plataforma o el tiempo que buscan documentación específica para poder resolver, datos que por su carácter personal, pueden ser abordados desde un paradigma cualitativo de investigación.

Cuando hablamos de aprender un lenguaje de programación, podemos ver que la metodología tradicional de evaluación del rendimiento estudiantil, se basa en analizar el “producto final” del código fuente desarrollado por los estudiantes. Pero, si tomamos como referencia a Seymour Papert (1980) podemos decir que una de las habilidades más importantes que los futuros programadores deben adquirir, no es tanto la habilidad de escribir código fuente, sino más bien, la capacidad de depurar y revisar el código escrito por ellos mismos o terceros.

Desde esa perspectiva, poder “documentar lo no documentado” (Rockwell, 2009) se convierte en una necesidad fundamental para ver los procesos de “prueba y error” en el desarrollo de código y la formación de una capacidad de detección temprana de errores sintácticos o semánticos. Podemos decir, a modo de hipótesis de trabajo, que cuando un estudiante escribe instrucciones en la computadora, a medida que avanza en su comprensión de lo que escribe, aumentará sus iteraciones de corrección sin necesidad de compilar o ejecutar en el intérprete su código fuente para poder detectar errores. Estas revisiones pueden ser analizadas viendo las pulsaciones de las teclas “suprimir”, “retroceder” o las interacciones con pulsaciones de los botones del mouse.

Pensar una actividad de investigación, implica tomar una postura epistemológica sobre cómo “vemos el mundo”, la forma en que accedemos e interpretamos los datos que recolectamos y la vigilancia epistemológica que debemos tener para poder considerar que nuestra pesquisa es lo suficientemente plausible para ser analizada y validada por nuestros colegas (Becker, 2019, p. 42). Desde esa perspectiva, pensar el trabajo de campo conlleva no solo prepararse para una recolección masiva de información, sin ningún tipo de criterio ni selección, por el contrario, es necesario posicionarse dentro de una teoría que nos permita cuál lente, “mirar” la complejidad del mundo de una forma que podamos operar sobre nuestro problema de investigación de forma metódica y relevante.

En una investigación de corte cualitativo, los procesos reflexivos y de relecturas de la información recolectada (entrevistas grabadas, videos, textos) forman parte fundamental del análisis que como investigadores llevamos a cabo para justificar los resultados a los que arribamos. Podemos decir que

metodológicamente, la legitimación del conocimiento desarrollado mediante alternativas de investigación cualitativa se realiza por la vía de la construcción de consensos fundamentados en el diálogo y la intersubjetividad (Sandoval Casilimas, 1996).

### **Una herramienta de laboratorio para la didáctica de programación**

La necesidad de poder contar con una herramienta que pudiera dar cuenta del proceso completo de trabajo de un estudiante y su entorno mientras aprende un lenguaje de programación, llevó al desarrollo una herramienta de laboratorio específica basada en la técnica de keylogging (Gilquin, 2019). Un keylogger, derivado del inglés: key “tecla” y logger “registrador”, lo que puede traducirse como “registrador de teclas”, es un tipo de software específico que se encarga de registrar las pulsaciones que se efectúan con el teclado, para posteriormente almacenarlas en un fichero o enviarlas a través de internet. Generalmente, su uso está asociado a técnicas de robo de información por parte de hackers (o crackers) interesados en obtener números de claves bancarias, password de accesos a sistemas remotos o cualquier otro tipo de información privada. No obstante la técnica de keylogging se muestra como una herramienta interesante para el análisis de laboratorio de prácticas estudiantiles (Lafuente, 2015).

El software desarrollado para el proceso de investigación se compone de dos herramientas: por un lado, tenemos un IDE (entorno de desarrollo integrado por sus siglas en inglés) minimalista, diseñado para enseñar a programar en lenguaje python de nombre **Torcaz**, y por el otro lado, un analizador de código y capturas de teclas escritas dentro del IDE, bautizado como **Burhus**.

El entorno de desarrollo está diseñado para grabar solamente las pulsaciones del teclado, el cambio de foco entre ventanas de software, la versión de código fuente que es ejecutada en el momento y los botones del mouse que se den dentro del espacio de trabajo, almacenando en un archivo CSV (valores separados por coma por sus siglas en inglés), donde guarda el tipo de evento generado, la fecha y hora (con minutos, segundos y milisegundos). Además, cada ejecución de código fuente es almacenada en un archivo .py por separado, junto con la salida estándar (stdout) y salida de errores (stderr) que genere el intérprete python cuando ejecute el código fuente escrito por los estudiantes. Por otro lado, el IDE tiene la capacidad de grabar en video el escritorio de la computadora (desktop screencasting) y el micrófono del laptop que se usará durante las actividades.

Un problema a la hora de hacer análisis cualitativo de grandes volúmenes de información, en video o textos, consiste en poder optimizar el tiempo de análisis que el investigador dedica a los datos recolectados. El gran volumen de datos que se recolecta de una sesión de trabajo de campo, obliga a tener un registro ordenado para poder analizar y obtener resultados útiles para un proceso de investigación de tipo cualitativo.



```

6 #####
7
8 #Import the library
9 #from midiutil.MidiFile import MIDIFile
10 import cv2
11 import numpy as np
12 from utilerias import mapFromTo
13 import apicaro
14 import time
15 # Create the MIDIFile Object
16 #MyMIDI = MIDIFile(1)
17
18 # Add track name and tempo. The first argument to addTrackName and
19 # addTempo is the time to write the event.
20 #track = 0
21 #time = 0
22 #MyMIDI.addTrackName(track,time,"Sample Track")
23 #MyMIDI.addTempo(track,time, 120)
24
25 # Add a note. addNote expects the following information:
26 #channel = 0
27 #pitch = 60
28 #duration = 10
29 #volume = 100
30 icaro = apicaro.puerto()
31 icaro.iniciar()
32
33 font = cv2.FONT_HERSHEY_COMPLEX
34 img = cv2.imread("/home/vbase/valentines/img2.jpg", cv2.IMREAD_GRAYSCALE)
35 _, threshold = cv2.threshold(img, 47, 5, cv2.THRESH_BINARY)
36 _, contours, _ = cv2.findContours(threshold, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
37 a=0
38 notas=[]
39 for cnt in contours:
40     approx = cv2.approxPolyDP(cnt, 0.01*cv2.arcLength(cnt, True), True)

```

*Ilustración 1: Entorno de desarrollo Torcaz*

Para poder analizar de forma eficiente este bagaje de información, es pertinente poder hacer una preselección que nos permita ordenar las prioridades de observación y análisis por parte del investigador. Para eso el software para la recolección de información mediante Keyloggin y screencapture, permite sincronizar la información obtenida de las pulsaciones de teclado y mouse así como el foco de la ventana activa, con el video resultante de grabar la captura de pantalla del escritorio. Con esa información, podemos determinar exactamente en qué momento el estudiante ejecutó el intérprete python, si la ejecución fue exitosa o dio error, cuando hizo correcciones antes de ejecutar al intérprete (apretando la tecla “backspace” o “delete”), también podemos ver cuanto tiempo conservó el foco en el IDE o fue a otros software (páginas WEB por ejemplo) o con análisis estadístico, ver en que momento tuvo mayor interacción de escritura por la cantidad de teclas presionadas en un momento dado. Esta información no nos permite dar cuenta de las complejas interacciones que los estudiantes hacen con las actividades propuestas, sin embargo, nos sirven como una guía orientativa para poder observar momentos específicos de la clase, mediante **Puntos de interés** que sirven para ahorrar tiempo de análisis.

Pensar en **puntos de interés** (PDI), orienta la investigación para poder hacer un análisis ordenado de los datos recolectados, de tal forma que si bien al ser una investigación de tipo cualitativa interpretativa, estos datos de recolección automático, permiten ahorrar tiempo de análisis así como generar una nueva instancia de información que puede resultar en nuevas interpretaciones del trabajo de los estudiantes.

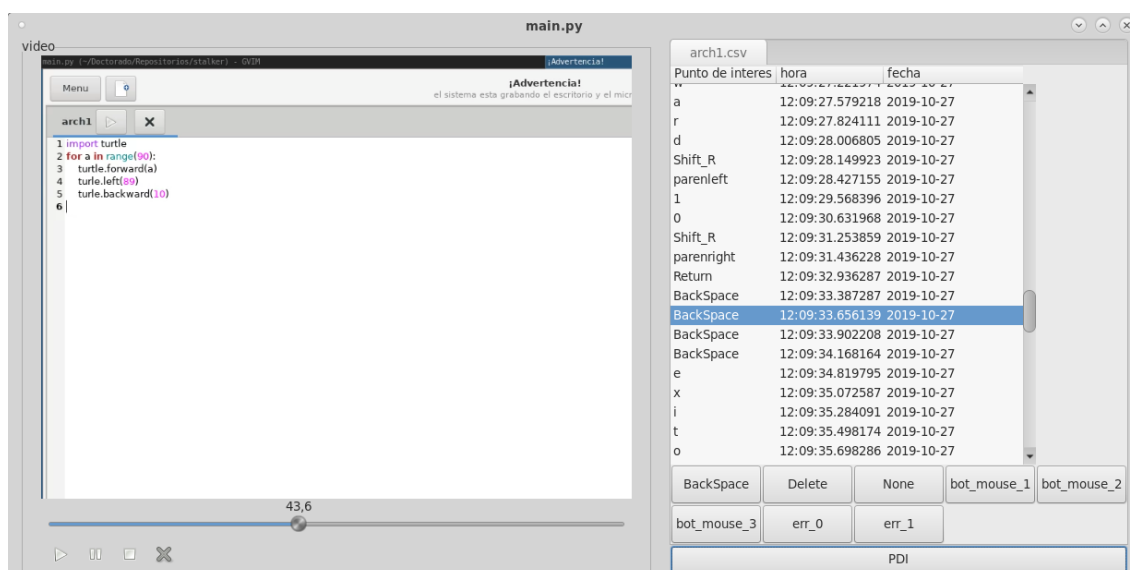
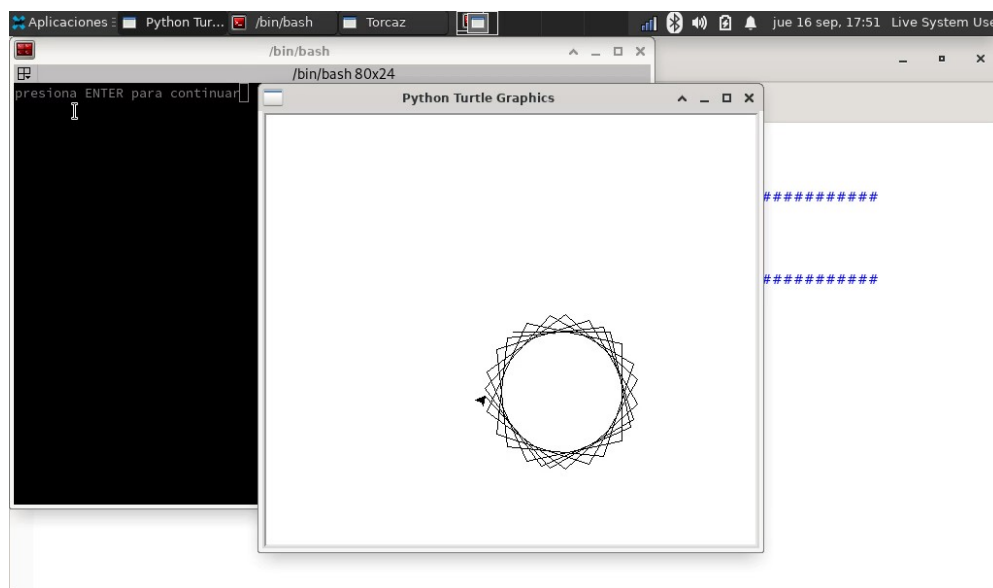


Ilustración 2: Analizador de código **Burrhus**

En ese sentido el diseño una herramienta analizadora de los proyectos grabados por **TORCAZ** Permitió orientar la visualización del trabajo de los estudiantes, así como la selección de los casos de estudio que fueron usados en el trabajo de campo. Gracias a la información recolectada, video del escritorio, registro de eventos y pulsaciones del teclado almacenado en CSV, así como las salidas estándar del sistema y código fuente escrito por los estudiantes, podemos hacer una preselección de la información a analizar y de esta forma ahorrar tiempo, dado que si bien la grabación de video da cuenta del trabajo ejercido en el aula por los estudiantes, existe momentos donde las actividades se “detienen”, como cuando el profesor está explicando una situación, o se está esperando que termine la ejecución de un código, o simplemente momentos donde los estudiantes dejan de escribir para pensar la situación o descansar de la tarea. Este “parate” en el flujo de trabajo, multiplicado por cada estudiante y por cada sesión, se vuelven muchas horas de grabaciones que no sirven en primera instancia para los efectos de investigación, y en ese sentido, la capacidad de filtrar y dar la opción para que el investigador pueda detectar automáticamente los momentos “interesantes” de observación, así como los perfiles de los estudiantes a analizar, se vuelve fundamental para un proceso de investigación enmarcado dentro del paradigma de investigación cualitativa interpretativa (Merlino, 2012; Serbia, 2007; Stake, 2013).



*Ilustración 3: Ejemplo de screencapture usando Torcaz*

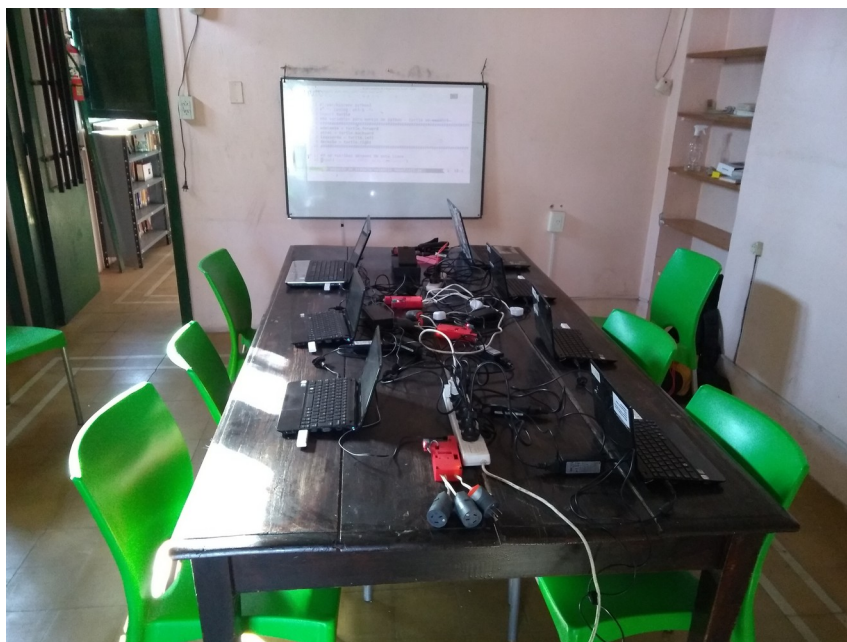
Desde esa perspectiva, el diseño de TORCAZ nos permite guiar la investigación cualitativa en función de un preanálisis de tipo cuantitativo, donde podemos ver los momentos de mayor escritura (mediante conteo de letras pulsadas en tiempos determinados), así como la cantidad de veces que se presiona teclas específicas (como la tecla Backspace). Podemos obtener, además, la cantidad de veces que se ejecutó el intérprete, con el resultado obtenido (ejecución correcta o con errores), o el tiempo de foco que tuvo cada ventana que se activó durante una sesión de trabajo. Esta información no pretende dar cuenta de la actividad elaborada por un estudiante, sino más bien orientar y guiar una selección de casos a estudiar, que completada con la observación de la clase en su formato más tradicional, nos permite seleccionar los casos de estudio de una forma metódica y ordenada, organizando el análisis posterior.

### **Trabajando con bibliotecas populares**

Torcaz y Burrhus fueron diseñados para trabajar como herramientas de trabajo de campo en la investigación doctoral sobre el uso de robótica educativa con software y hardware libre en espacios no formales de educación, usando sistemas de domótica educativa (Realinho, 2015). El trabajo de campo se ordenó en función de las condiciones impuestas a raíz del ASPO y las regulaciones imperantes con respecto a la distancia social en los cursos escolares desde el mes de septiembre del 2021 a octubre del mismo año. Teniendo en cuenta esas limitantes y entendiendo que un curso de programación que utilice sistemas electrónicos, no puede ser extrapolado a una modalidad virtual de forma completa, se tomó la decisión de generar un formato de “burbujas” con cantidades reducidas de participantes, de forma de minimizar el contacto entre los estudiantes que trabajarían de forma individual, un estudiante por computadora y dispositivo electrónico de domótica, teniendo un máximo de 6 estudiantes por comisión y 2 horas de clase. La estructura curricular fue determinada por los contenidos a abordar en un curso inicial de programación, de corta duración y en un espacio no formal de trabajo. Desde esa perspectiva se optó por organizar talleres semanales con 4

encuentros semanales (1 mes completo), donde se ordenó la agenda para poder trabajar con 3 bibliotecas populares de forma concurrente y en paralelo y así poder optimizar el uso de computadoras que fueron entregadas en préstamo por el FAMAFA.

Las bibliotecas populares con las que se trabajó fueron la Biblioteca Popular de Alberdi, la biblioteca popular Brújula Barrial y la biblioteca popular Juana Manuela Gorriti.



*Ilustración 4: Curso impartido en la biblioteca popular de Alberdi (Córdoba, Argentina)*

Trabajar con bibliotecas populares implica una serie de decisiones, tanto metodológicas como logísticas, en función de la heterogeneidad general que suelen tener las mismas, dado que al ser agrupaciones civiles más o menos formales (Dobra & Ríos, 1999), las herramientas, infraestructura y esquemas organizacionales son construidas y sostenidas por las personas que integran la biblioteca en un momento particular.

Así podemos ver que para trabajar con tecnologías digitales, las bibliotecas en las que se participó, estaban en situaciones muy disímiles en función de sus capacidades. La Biblioteca Popular de Alberdi contaba con un laboratorio de computadoras, de aproximadamente unos 10 años de antigüedad, que eran mantenidas por un miembro de la biblioteca que había conseguido donaciones de equipos y muebles para las mismas. Sin embargo, con su fallecimiento, el laboratorio dejó de tener mantenimiento y fue quedando obsoleto e inutilizable, al punto que fue desarmado para re aprovechar el espacio físico para otras actividades, entre ellas el curso de domótica.

En el caso de la biblioteca “La brújula Barrial”, el panorama era más complejo, dado que solo contaban con un espacio físico general, donde dan talleres de manualidades y sostienen un



“roperito” de venta de ropa usada, para financiar gastos operativos de la casa donde funciona la biblioteca, sin tener computadoras, aunque podían prestar un proyector y red de wifi a los efectos del curso que se impartió en el lugar.



*Ilustración 5: Biblioteca popular "La Brujula Barrial"*

La biblioteca Juana Manuela Gorriti se encontraba en un proceso de mudanza de su sitio original a una nueva instalación y por eso su infraestructura estaba más limitada, contando únicamente con un espacio con mesas, wi-fi y electricidad.

En general podemos decir que solamente se pudo contar con espacio físico, electricidad e internet a través de wi-fi, dado que los talleres que utilizan computadoras son muy poco común en estos espacios, basta hacer una pesquisa en los anuarios que genera la CONABIP (Comisión Nacional de Bibliotecas Populares) contando los trabajos efectuados por las distintas bibliotecas populares a lo largo del país para ver que hay muy poca oferta de cursos donde se usan las TICs.

Las tres bibliotecas elegidas para esta instancia de investigación tienen una historia diferente. La Biblioteca Popular de Alberdi es una institución con 109 años de antigüedad, que forma parte del CONABIP y recibe un “subsido de subsistencia”, con un catálogo de libros de cultura general extenso y una función de biblioteca más tradicional, establecida en el barrio de Alberdi con un edificio propio y compuesta por una Asociación civil que controla y administra a la biblioteca.

La biblioteca Juana Manuela Gorriti no forma parte de la CONABIP, pero sí de la FEBIPO (Federación de Bibliotecas Populares), dado que al ser una biblioteca especializada en literatura feminista, no cumple con los requisitos impuestos por la CONABIP y, por tanto, no recibe ni subsidios económicos ni materiales por parte del organismo estatal.



*Ilustración 6: Biblioteca popular Juana Manuela Gorriti*

Por último, La biblioteca popular “La Brújula Barrial” tiene sus comienzos en el año 2001, donde una agrupación política barrial, decide instituirse como Biblioteca para tener una interacción territorial en el barrio de Acosta, con un fuerte énfasis en la interacción con las escuelas de la zona, así como desarrollo de actividades para la comunidad. En la entrevista con Lili Robledo (directora de la biblioteca), ella comenta que eligieron el modelo de Fundación para constituir la biblioteca porque al ser un tipo de personería jurídica con una organización vertical, que no permite la entrada de nuevos miembros en la comisión directiva sin la autorización de los miembros de la misma, evitando de esta forma el ingreso de “punteros políticos” que podían “desvirtuar” los valores por los que se creó la biblioteca. El hecho de ser fundación, impidió a la Brújula Barrial formar parte de la CONABIP, y la casa donde funciona la biblioteca fue adquirida con fondos de la fundación Arcor.

Al tener realidades tan disímiles y en general una escasa infraestructura computacional que permitiera dar cuenta de un taller de programación con domótica educativa, llevó a diseñar un sistema que permitiera trasladar todo el equipamiento necesario para poder establecer los cursos de programación que se implementarían en las distintas bibliotecas. Para eso se procedió a conseguir el siguiente equipamiento:

- 6 Netbooks ASUS de 9 pulgadas (entregadas en préstamo por el FAMAF).
- 12 pen drives USB con sistemas GNU/Linux instalados y pre-configurados para dar los cursos.
- 6 placas para enseñanza de programación basadas en domótica educativa (Domoticaro) ensambladas y configuradas.

- 1 fuente de alimentación eléctrica de 12 volt con 6 cables de conexión para las placas.
- 6 dispositivos eléctricos como luces LEDs de colores, bombas de agua de 12 Volt y ventiladores tipo cooler de 12 Volts.
- 2 zapatillas eléctricas de 6 bocas cada una, así como un alargador de 15 metros.
- 1 mochila de campamento para poder trasladar todo el equipamiento de forma fácil a las distintas bibliotecas.

La implementación de un “kit viajero” permitió efectivamente encarar los cursos de programación en las distintas bibliotecas, dado que ninguna de las 3 tenía una infraestructura técnica capaz de dar cuenta de las necesidades que se tendrían en los cursos propuestos.

Sin embargo, esto agregó la complejidad de armar y desarmar el laboratorio cada vez que se tenía que hacer el curso en cuestión, agregando tiempo de trabajo antes y después de cada clase, así como la imposibilidad de dejar equipos instalados (luces por ejemplo) en las instalaciones. Este ordenamiento generó algunas decisiones metodológicas con respecto al dictado de los cursos, por un lado, los chicos no podrían llevarse los equipos a sus casas para estudiar, y no se pudo implementar un proyecto integrador que permitiera usar las placas para automatizar los espacios comunes de la biblioteca.

Teniendo en cuenta estos limitantes, sumados a los protocolos de distanciamiento social, se decidió crear una convocatoria para los cursos con un máximo de 6 estudiantes por cohorte, teniendo cuidado de ocupar la agenda semanal de tal forma que permita el trayecto desde una biblioteca a otra con el equipamiento y la preparación de los espacios de trabajo.

Los cursos fueron organizados en 4 encuentros semanales, de 2 horas de duración cada uno con un máximo de 6 participantes por cursos. De esta forma los estudiantes usaron el hardware diseñado para el curso, que constó de un kit de electrónica para control de domótica mediante las placas diseñadas para ese fin, además de tener un pendrive con el sistema operativo GNU/Linux ya configurado con el software de investigación **Torcaz** (screencapture + keylogger) que permitió hacer un seguimiento del trabajo de los estudiantes.

El curso fue diseñado de forma que cada estudiante tuviera una netbook y una placa de domótica para su uso individual, sumando algún dispositivo eléctrico como luces navideñas, de colores, un ventilador o una mini bomba de agua adquiridos con el proyecto CONSOLIDAR Secyt UNC dirigido por la Dra. Luciana Benotti. Los estudiantes trabajaron con el software de investigación **Torcaz**. Si bien el curso está diseñado para enseñar programación a estudiantes secundarios, también es una instancia de investigación, por tanto, se buscó poder tener una herramienta que permitiera hacer una recolección ordenada de datos para poder analizarla. En ese sentido, el diseño de la herramienta **Torcaz** permitió tener un registro ordenado de las actividades que los estudiantes

fueron desarrollando durante el cursado, y de esa manera poder analizar las capturas de pantalla creadas por el software.

En la Biblioteca “la Brújula Barrial”, la convocatoria de estudiantes fue llevada a cabo mediante carteles pegados en la puerta de la biblioteca, así como folletos entregados en escuelas de la zona y una convocatoria interna con los participantes del espacio. Al final, los inscriptos fueron 4 estudiantes, aunque participaron 3 en la totalidad del curso, que se dio en el mes de septiembre del año 2021, los días jueves de 4 Pm a 6 Pm.

En el caso de la Biblioteca Popular de Alberdi, la convocatoria fue llevada a cabo exclusivamente por internet, usando la plataforma Facebook y mediante mensajes de WhatsApp, en total se inscribieron 12 personas, lo que llevó a dividir las clases en dos turnos, para poder mantener un espacio de burbuja adecuado en función del lugar asignado para el trabajo. De los 12 estudiantes inscriptos, completaron el curso 8 estudiantes, teniendo 4 estudiantes a la mañana (sábados de 10 Am a 12 Am) y 4 estudiantes en el turno tarde (sábados de 4 Pm a 6 Pm), durante el mes de septiembre del año 2021.

La biblioteca popular Juana Manuela Gorriti, al momento de la convocatoria para el curso, se encontraba en plena mudanza de la casa donde estaban a un nuevo espacio en la zona del abasto, eso implicó tener muy poca interacción zonal, lo que conllevó a hacer una convocatoria exclusivamente por redes sociales, teniendo como resultado la inscripción de 2 estudiantes.

El uso de una herramienta de investigación como la presentada, permitió analizar caso por caso la participación de los estudiantes de cada biblioteca donde se hizo la experiencia. Esta cantidad de información posibilitó hacer un análisis en profundidad de las prácticas de los estudiantes, así como de su interacción con la computadora, sus formas de aprender y aprehender conceptos de programación así como las interacciones que efectuaban entre ellos. Por ejemplo, durante una de las clases impartidas, se le pidió a los estudiantes que copiaran una serie de código fuente para poder aprender la sintaxis básica de la API necesaria para poder interactuar con el hardware de domótica que se usó en los cursos. Esta práctica, que en principio no tuvo gran interés a nivel de investigación, nos permitió probar algunas herramientas de análisis tomando el archivo .CSV con las teclas pulsadas por los estudiantes y buscando patrones que se pudo luego examinar de forma cualitativa en los videos grabados por Torcaz. En ese sentido se analizó los distintos archivos .CSV con PYTHON buscando la incidencia del uso de algunas teclas como “backspace” o “delete”, entendiendo que una de hipótesis planteadas pasa por considerar que un programador que autocorrije su código antes de necesitar enviarlo al intérprete para que esté de un error, se lo puede considerar un programador que comprende lo que está escribiendo y, por tanto, podemos considerarlo un programador avanzado (Papert, 1987, p. 38).

Sin embargo, cuando se analizó los distintos archivos .CSV, apareció que una estudiante tenía la mayor tasa de incidencias de la tecla “backspace” con respecto a sus compañeros. Si bien esta información por sí solo no dice nada interesante, si permitió orientar un examen minucioso de la

clase grabada en la computadora de este estudiante para tratar de visualizar la actividad que estaba haciendo y que llevaba a una gran cantidad de corrección de su código fuente.

Cuando se analizó el video de la clase se encontró que el estudiante si cometía un error de sintaxis o el intérprete marcaba un fallo, borraba todo el código hasta donde estaba la revisión y lo volvía a escribir, en vez de usar las flechas del teclado. Por eso en la segunda clase se la entrevistó para averiguar que tipo de actividad hacía cuando escribía. Resultó que no sabía utilizar un teclado de computadora, y su interacción era más parecida a la que tenía con su celular, borrando todo lo que escribe cuando encuentra un error. Luego de esta pequeña entrevista, resultó notable ver la diferencia entre estudiantes que tenían o habían tenido una computadora personal para sus estudios y los que solo interactúan a través de sus dispositivos celulares. Esta divergencia llegó al punto que se necesitó hacer una pequeña intervención explicando el uso básico de atajos de teclado y manejo de un editor de texto para poder seguir con la dinámica de la clase propuesta.

### **Consideraciones finales**

La utilización de un análisis cuantitativo no pareciera a priori útil para poder sacar conclusiones sobre los sistemas de actividades creados por los estudiantes, pero sí permitió obtener algunos datos útiles para poder orientar una segunda etapa de análisis más específico (Serbia, 2007).

Podemos decir que contar con las clásicas filmaciones de toda el aula, cuadernos de campo donde el investigador anota sus observaciones y además con grabaciones pormenorizadas de las interacciones de los estudiantes en sus computadoras, conservando la mayor cantidad de información creada por ellos, es una herramienta poderosa para poder estudiar los procesos cognitivos que los participantes de un curso de programación ponen en juego cuando aprenden a programar (Grijalba et al., 2018). Poder no solo ver el producto final desarrollado por los estudiantes (Garzón Morales, 2015), sino tener todas las iteraciones llevadas a cabo (con manejo de pruebas y error), así como poder contar con grabaciones de el escritorio de las computadoras de los estudiantes, permite tener herramientas para analizar de forma cualitativa las actividades de los estudiantes y dar cuenta de los procesos de aprendizaje que están interviniendo.

## Referencias Bibliográficas

- Becker, H. (2019). *Datos, pruebas e ideas: Por qué los científicos sociales deberían tomárselos más en serio y aprender de sus errores*. Siglo XXI Editores.
- Botella Nicolás, A. M., & Ramos Ramos, P. (2019). Investigación-acción y aprendizaje basado en proyectos. Una revisión bibliográfica. *Perfiles educativos*, 41(163), 127-141.
- Dobra, A. T., & Ríos, D. R. (1999). Las bibliotecas populares argentinas. *Métodos de Información (MEI)*, 6(32-33), 68-72.
- Garzón Morales, N. E. (2015). *Una propuesta metodológica para la enseñanza de programación en ingeniería multimedia*.
- Gilquin, G. (2019). Screencasting and keylogging as pedagogical tools to enhance writing skill development. *27th EUROCALL conference*.
- Grijalba, W. A., Castillo, J. A., Campos, R. A., & Estrella, P. (2018). Process-based assessment of computer science students. *2018 XIII Latin American Conference on Learning Technologies (LACLO)*, 363-370.
- Lafuente, R. A. (2015). *Keylogging para el estudio de los procesos cognitivos del traductor* [B.S. thesis].
- Merlino, A. (2012). *Investigación cualitativa y análisis del discurso: Argumentación, sistemas de creencias y generación de tipologías en el estudio de la producción discursiva*. Biblos.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.
- Papert, S. (1987). *Desafío de la mente*. Ediciones Galapagos.
- Realinho, V. (2015). Low Cost Domotic System based on Open Hardware and Software. *The Eighth International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services*.
- Rockwell, E. (2009). La experiencia etnográfica. *Historia y cultura en los procesos educativos*,
- Rodríguez-Sandoval, E., Vargas-Solano, É. M., & Luna-Cortés, J. (2010). Evaluación de la estrategia "aprendizaje basado en proyectos". *Educación y educadores*, 13(1), 13-25.
- Sandoval Casilimas, C. A. (1996). *Investigación cualitativa*.

Serbia, J. M. (2007). Diseño, muestreo y análisis en la investigación cualitativa. *Hologramática*, 4(7), 3.

Stake, R. (2013). Estudios de casos cualitativos. *Las estrategias de investigación cualitativa*.

Zapata-Ros, M. (2015). Analítica de aprendizaje y personalización. *Campus virtuales*, 2(2), 88-118.

# Evaluación de calidad de Entornos de Desarrollo Integrado de Robótica Educativa

Martín LOBOS, Silvia BAST, Gustavo ASTUDILLO

GrIDiE, Departamento de Matemática, FCEyN, UNLPam

lobmar146@gmail.com, {silviabast, astudillo}@exactas.unlpam.edu.ar

## Resumen

Este artículo es la continuación de una investigación precedente enfocada en la evaluación de Entornos de Desarrollo Integrado (EDI) de Robótica Educativa. El objetivo es identificar el EDI que mejor se adapta a una propuesta didáctica de un Taller para ingresantes que hace uso de robótica educativa. Se propuso realizar una evaluación de los productos de software utilizando la estrategia denominada *Goal oriented context aware measurement and evaluation*. La investigación avanzó en el desarrollo del modelo de calidad y en la definición de métricas para cada uno de los atributos del modelo.

En el presente trabajo se continúa sobre el proceso de medición, la definición de indicadores, evaluación y los resultados finales de la investigación. Se presenta, además, el ranking de los IDE obtenido luego del proceso de medición.

Palabras clave: Entorno de desarrollo integrado, Medición, Evaluación, GOCAME, ISO 25010, Robótica Educativa.

## Introducción

Los integrantes del Grupo de Investigación GriDiE en conjunto con el equipo de la cátedra Introducción a la Computación (IC) de la Facultad de Ciencias Exactas y Naturales (FCEyN) de la Universidad Nacional de La Pampa (UNLPam), trabajan en propuestas de enseñanza que permitan acercar a los estudiantes que ingresan a la carrera de Profesorado en Computación (PUC) a los conceptos básicos de la programación.



Una de las propuestas del equipo fue el Taller de Introducción a la Programación (TIP) que se desarrolla en el marco del curso de ambientación para ingresantes en formato semipresencial.

Los destinatarios del TIP son los estudiantes que deben cursar IC y que, generalmente, no han tenido contacto previo con la programación. El TIP hace uso de la programación en bloques (Astudillo et al, 2016), El taller incluye los conceptos de: alternativa condicional, repetición y nociones de variables a través de una propuesta que hace uso de la robótica educativa (Astudillo et. al, 2019) y se sustenta en los principios del construccionismo (Papert, 1990), el buen aprendizaje (Pozo, 2008) y aprendizaje basado en problemas (Barrows, 1986).

La decisión de incluir robótica educativa se fundamenta en que los robots resultan la conexión ideal entre la programación con una impronta lúdica y la representación de las instrucciones en un contexto real, en el que los estudiantes pueden participar activamente construyendo aprendizajes a partir de la experiencia (Monsalves González, 2011; Ruiz-Velasco, 2007).

Para lograr la efectiva aplicación de la secuencia didáctica haciendo uso de robótica educativa fue necesario: definir un kit (sensores y actuadores), seleccionar un EDI, al que se sumó además un simulador (para tareas extra-clase). También se desarrollaron un conjunto de ejercicios (actividades guiadas por el docente donde los estudiantes se apropian de nuevos conceptos) y problemas (actividades sin guía del docente donde los estudiantes ponen en práctica dichos conceptos).

La presente investigación es una continuación del trabajo previo (Lobos et. al, 2021) y se lleva a cabo con el objetivo de identificar el EDI que mejor se adapte a la propuesta didáctica que se aplica en el TIP.

El artículo se organiza como sigue: en la sección Marco teórico se exponen las bases conceptuales que dan sustento a la presente investigación; seguidamente, se presenta el Contexto de aplicación (que refiere al trabajo previo), para luego avanzar sobre el Proceso de medición, la Definición de los indicadores y el Proceso de evaluación del que surgen los resultados. Finalmente, aparecen las Conclusiones y trabajos futuros.

## Marco teórico

El hecho de incluir un EDI en el taller sumado a la diversidad de productos de este tipo que se presentan en el mercado, motiva la presente investigación.

Para llevar a cabo el proceso de evaluación de calidad de los EDI se hizo uso del estándar que establece la norma ISO 25010:2011 para productos de software. En la mencionada norma se define

la calidad de un sistema como “el grado en que el producto satisface los requisitos declarados e implícitos de los diversos interesados y, por lo tanto, proporciona valor”.

Para medir la calidad de un producto de software, se hace necesario construir modelos que permitan establecer primeramente las características que se desean evaluar, para luego descomponerlas en factores de manera jerárquica (Kitchenham & Pfleeger, 1996).

Para llevar adelante el proceso se usó la estrategia de medición y evaluación denominada *Goal Oriented Context Aware Measurement and Evaluation* (GOCAME) (Olsina et al, 2008).

Según GOCAME, la primera actividad del proceso consiste en especificar la necesidad de información, y establecer los requerimientos a partir de los cuales surge el modelo de calidad que presenta una estructura de tipo jerárquica. El mencionado modelo toma sus bases del estándar ISO 25010:2011 “Calidad del producto de Software” (Figura 1) que presenta 8 características. De cada una de las mismas, se desprenden a su vez, grupos de sub-características.



Figura 1. Modelo de calidad del producto software de ISO/IEC 25010. Imagen extraída de iso.25000.com

Cada una de las sub-características del modelo puede desglosarse en un conjunto de subconceptos y atributos que se generan de acuerdo con los requerimientos discutidos y establecidos por el grupo de investigación.

El siguiente paso consiste en definir las métricas, para posteriormente aplicarlas en el proceso de medición. Una métrica representa un mapeo de un atributo a una variable que toma valores categóricos o numéricos (Olsina, 2008). Para la definición de las mismas se debe indicar el método de medición (Objetivo/Subjetivo), la escala (nominal, ordinal, intervalo, proporción y absoluta) y el tipo de valor. Las métricas pueden ser directas (no depende de otra métrica) o indirectas (dependen de otra métrica).

Luego, se definen los indicadores que incluyen la especificación del procedimiento de cálculo y la escala definidos, además del modelo y los criterios de decisión, con el objetivo de proporcionar una estimación o evaluación de un concepto calculable con respecto a la necesidad de información

definida (Covella, 2005). Finalmente, sigue el proceso de evaluación, que es la actividad en la que se usa la definición de los indicadores para producir un valor (Covella, 2005). Con los resultados de la evaluación es posible realizar el análisis de la información obtenida y realizar las recomendaciones pertinentes de acuerdo con la necesidad de información.

## Contexto de aplicación

En una investigación previa (Lobos et. al, 2021) se comenzó con el proceso de medición de calidad de EDI para robótica educativa, con el objetivo de determinar cuál o cuáles cuentan con la mayor flexibilidad para adaptarse a la propuesta didáctica del TIP. Fueron seleccionados, para su evaluación, un total diez (10) IDE que utilizan bloques para la programación y se enfocan en robótica educativa:

1. Mixly
2. Visualino
3. Ardublockly
4. BlocklyDuino
5. Blockly@rduino
6. Blockly4Arduino
7. ArduinoBlocks
8. Facilino (Offline)
9. Facilino (Online)
10. AeLE

Se inició el proceso de medición de calidad de software estableciendo la necesidad de información (el objetivo que guía la investigación), y a partir de la misma se definieron los requerimientos tomándose como base el estándar ISO 25010:2011 del que se seleccionaron tres características (Figura 2) que resultan de relevancia para alcanzar el objetivo planteado. A continuación, se generó el modelo de calidad (disponible en [https://gridie.exactas.unlpam.edu.ar/?page\\_id=115](https://gridie.exactas.unlpam.edu.ar/?page_id=115)) con atributos discutidos y establecidos por el grupo de investigación. Seguidamente, para cada uno de los atributos del modelo se definieron las métricas.

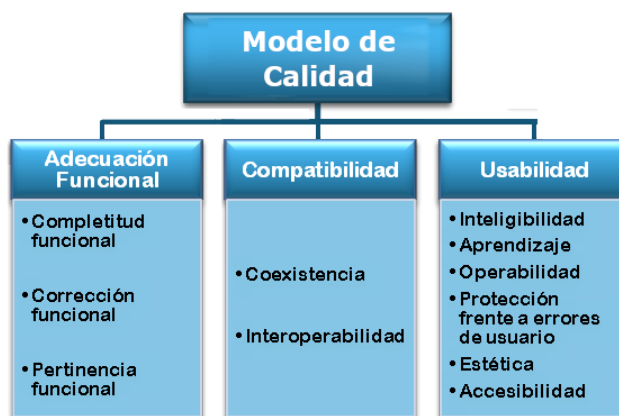


Figura 2. Ilustración del modelo de calidad generado con sus características y subcaracterísticas

## El proceso de medición

A partir de la definición de las métricas, se lleva a cabo el proceso de medición de cada uno de los EDI, que consiste en aplicar para cada atributo del modelo de calidad, el método de medición y la escala definidos.

En la Tabla 1 puede observarse una parte de los resultados del proceso de medición. El modelo completo se encuentra disponible en: [https://gridie.exactas.unlpam.edu.ar/?page\\_id=115](https://gridie.exactas.unlpam.edu.ar/?page_id=115).

	Entornos de Desarrollo		
	Mixly	Visualino	Ardublockly
1. Adecuación Funcional			
1.1 Completitud Funcional			
1.1.1. ¿Presenta aplicación móvil?	0	0	0
1.1.2. ¿Se puede usar en línea?	No	No	Si
1.1.3. ¿Presenta uso desconectado?	Si	Si	Si
1.2. Correctitud Funcional			
1.2.1. ¿ Autocompleta huecos en los bloques que necesitan la inserción de valores/datos/expresiones?	Si	No	Si
1.2.2 ¿Presenta listas desplegadas en los espacios de los bloques que pueden tomar distintos valores?	2	2	2
1.2.3. ¿Incluye un espacio para la utilización de la sentencia delay() en los bloques que necesitan su uso para su correcto funcionamiento?	1	2	0
1.2.4. ¿Los bloques son representativos respecto a la instrucción/funcionalidad?	1	3	3

1.2.5. ¿Presenta mensajes y warnings cuando se producirá un error de lógico, compilación o ejecución al posicionar los bloques?	No	No	No
...			
<b>2.2 Estética e Interfaz de Usuario</b>			
2.2.1. ¿Presenta categorías de bloques distinguidas con imágenes?	Si	No	No
2.2.2. ¿Los botones representan correctamente su funcionalidad?	100,00	100,00	100,00
2.2.3. ¿Permite cambiar el formato de los bloques?	No	No	No
2.2.4. ¿Presenta categorías y subcategorías de bloques con distinciones estéticas?	Si	Si	No
2.2.5. ¿Presenta un adecuado contraste entre la letra y el fondo?	37,5	33,33333333 3	54,5454545 5
2.2.6. ¿Los bloques de sensores y actuadores cuentan con una imagen ilustrativa del dispositivo?	0	1	0

Tabla 1. Ejemplo de la tabla de medición con alguno de los atributos (producción propia)

Como se puede observar en la Tabla 1, para el atributo 1.1.3. *¿Presenta uso desconectado?*, se utilizan los tipos de valores (Si, No), con representación discreta y escala nominal.

Para alguno de los atributos el proceso de medición fue simple, ya que requirió observar características sencillas como la mencionada anteriormente. Sin embargo, en otros casos fue necesario realizar un proceso de medición exhaustivo y detallado. Por ejemplo, el atributo 2.2.5 *¿Presenta un adecuado contraste entre la letra y el fondo?*, tiene asociada una métrica indirecta, lo que implica que depende de otra para ser medida. Por lo tanto, en primer lugar se tuvo que medir la métrica *Cantidad total de colores de bloques del IDE*, que consiste en contar todos los colores. Para ello se hizo uso del Colorpicker<sup>1</sup> que permitió extraer los valores de los colores que son utilizados para los bloques del EDI. Esto requirió recorrer todas las categorías y dentro de cada una de ellas analizar bloque a bloque. Una vez obtenidos todos los colores se pudo pasar a medir el atributo. Esta actividad requirió utilizar la página web WebAIM Contrast Checker<sup>2</sup> para comparar color a color, con el objetivo de ver si aplicaba para el atributo contraste mínimo del estándar WCAG 2.0<sup>3</sup>. Finalmente, tomando como datos la cantidad de colores que pasaban la prueba y el total obtenido anteriormente, se calcula el valor en porcentaje que la métrica requería.

<sup>1</sup> Colorpicker es una desktop app para la fácil extracción de colores. <https://colorpicker.fr/>

<sup>2</sup> WebAIM Contraste Checker es una web de accesibilidad para el chequeo de contrastes entre colores de fondo y texto aplicando el estándar WCAG 2.0. <https://webaim.org/resources/contrastchecker/> obtenidos

<sup>3</sup> Se puede acceder al estándar completo en: <https://www.w3.org/TR/WCAG20/>

Este proceso fue similar para las métricas indirectas que requerían generar un valor porcentual, tales como 2.2.2. *¿Los botones representan correctamente su funcionalidad?*, entre otras.

## La definición de indicadores

Una vez finalizada la actividad de medición, se continúa con el proceso de evaluación, que consiste en tomar como insumo las medidas para generar información contextualizada. Para ello se desarrollaron los indicadores que señalan qué valores son aceptables y qué valores no. Existen tres (3) tipos de indicadores: Globales, Parciales y Elementales. Los indicadores elementales son los que responden a cada uno de los atributos y se definen en base a estos. Por ejemplo, para el atributo 1.1.3. *¿Presenta uso desconectado?* que se mide utilizando los tipos de valores (Si,No), la tarea de los indicadores es definir cuál de los dos valores es satisfactorio y cuál no, ya que la ausencia o presencia de alguna característica puede ser satisfactorio o no según el atributo. A continuación, un ejemplo de la especificación de Indicador, tomando este atributo como ejemplo.

- **Indicador elemental del tipo 1 (IE1):**
- **Modelo Elemental:**
  - *Especificación: el mapeo es  $IE1 = 100$  si  $MD = Si$ ; es  $IE1=0$  si  $MD=$  es No, donde  $MD$  es una métrica directa con las siguientes características:*
    - *Representación: Discreta*
    - *Tipo de Valor: (Si, No)*
    - *Tipo de Escala: Nominal*
- **Criterio de Decisión: [2 niveles]**
  - *Nombre 1: Satisfactorio si  $IE1=100$* 
    - *Descripción: Indica la presencia de la característica, funcionalidad o propiedad medida.*
  - *Nombre 2: Insatisfactorio si  $IE1=0$* 
    - *Descripción: Indica la ausencia de la característica, funcionalidad o propiedad medida.*

Como se puede observar, la especificación del indicador mapea a un valor numérico que, en este caso puede ser 0 o 100. A continuación, se describe la representación, tipo de valor, tipo de escala y finalmente se definen los criterios que indicarán qué valores se consideran aceptables o

satisfactorios y cuáles no. En este caso para el valor **Sí** es Satisfactorio asignándole un valor de 100 y para **No** es insatisfactorio y se le da un valor de 0.

En total, se definieron seis (6) indicadores elementales que abarcan todos los tipos de valores obtenidos durante la medición. Se puede encontrar el documento con todos los indicadores accediendo a [https://gridie.exactas.unlpam.edu.ar/?page\\_id=115](https://gridie.exactas.unlpam.edu.ar/?page_id=115).

A partir de los indicadores elementales se definen los Globales y Parciales, que corresponden a las categorías que engloban o incluyen atributos. Para estos se aplicará la siguiente fórmula:

$$\text{Indicador} = \Sigma (\text{Peso} \times \text{Indicador elemental}).$$

Donde *Peso* es el nivel de importancia que se le asignará según la necesidad de información de la investigación. Por lo tanto, en la asignación de los pesos, se busca reflejar qué características de un EDI resultan más relevantes para la investigación. La importancia de este proceso radica en que los resultados que se obtendrán están directamente ligados a los pesos que se asignen a los indicadores. A modo de ejemplo, se muestra en la *Tabla 2* los indicadores parciales y globales para la característica de adecuación funcional.

<b>1. Adecuación Funcional</b>	<b>= 0,33 * 1.1. + 0,33 * 1.2. + 0,33 * 1.3.</b>
<b>1.1 Completitud Funcional</b>	<b>=0.3* 1.1.1 + 0.2 * 1.1.2 + 0.5* 1.1.3</b>
1.1.1. ¿Presenta aplicación móvil?	Indicador Elemental tipo 3 (IE3)
1.1.2. ¿Se puede usar en línea?	Indicador Elemental tipo 1 (IE1)
1.1.3. ¿Presenta uso desconectado?	Indicador Elemental tipo 1 (IE1)

*Tabla 2. Indicadores Parciales y Globales con sus respectivos pesos. En amarillo, se indican los pesos asignados (producción propia).*

En el indicador parcial *1.1. Completitud funcional*, se le da el peso de 0,5 al indicador elemental del atributo *1.1.3. ¿Presenta uso desconectado?* ya que esta información es sumamente importante por el piso tecnológico disponible en las escuelas secundarias de La Pampa donde la conectividad es escasa y en algunos colegios directamente nula, tornándose así, el uso desconectado como una característica de alto peso. De esta manera, se fueron estableciendo los pesos para cada indicador de acuerdo a la relevancia que presentan para la necesidad de la información.

## El proceso de Evaluación

Finalmente, se lleva a cabo el proceso de evaluación, aplicando los indicadores. Los resultados que se han obtenido serán los que determinen cuál EDI será recomendado. Como primer paso, en base a los resultados arrojados por las mediciones, se obtienen los valores de los indicadores elementales que corresponden a cada atributo. Luego se calculan los indicadores parciales y globales aplicando la fórmula descrita anteriormente.

	Mixly	Visualino	Ardublockly
<b>TOTAL</b>	70,9536	65,6158	63,4301
<b>1. Adecuacion Funcional</b>	66,726	65,868	67,386
<b>1.1 Completitud Funcional</b>	80	80	100
1.1.1. ¿Presenta aplicación móvil?	100	100	100
1.1.2. ¿Se puede usar en línea?	0	0	100
1.1.3. ¿Presenta uso desconectado?	100	100	100
<b>1.2. Correctitud Funcional</b>	49,8	53,2	53,2
1.2.1. ¿Autocompleta huecos en los bloques que necesitan la inserción de valores/datos/expresiones?	100	0	100
1.2.2. ¿Presenta listas desplegadas en los espacios de los bloques que pueden tomar distintos valores?	66	66	66
1.2.3. ¿Incluye un espacio para la utilización de la sentencia delay() en los bloques que necesitan su uso para su correcto funcionamiento?	50	100	0
1.2.4. ¿Los bloques son representativos respecto a la instrucción/funcionalidad?	33	100	100
1.2.5. ¿Presenta mensajes y warnings cuándo se producirá un error de lógico, compilación o ejecución al posicionar los bloques?	0	0	0
<b>1.2. Pertinencia Funcional</b>	49,8	53,2	53,2

Tabla 3. Ejemplo de mediciones reemplazadas por los valores otorgados por los indicadores elementales y resultados de indicadores parciales, globales y resultado final. Con fondo gris, se denotan los indicadores parciales. Con verde el indicador global 1. Adecuación Funcional. (producción propia)

La Tabla 3 presenta un recorte de la tabla de evaluación (la medición completa está disponible en [https://gridie.exactas.unlpam.edu.ar/?page\\_id=119](https://gridie.exactas.unlpam.edu.ar/?page_id=119)), en este ejemplo, se puede ver cómo los atributos 1.1.1. ¿Presenta aplicación móvil?, 1.1.2. ¿Se puede usar en línea? y 1.1.3. ¿Presenta uso desconectado? A partir del resultado obtenido en la medición se ha obtenido el valor del indicador elemental. Estos tres atributos corresponden al indicador elemental que mapea



para aquellas métricas con solo dos (2) opciones, se toma como Satisfactorio el Si y como Insatisfactorio el No  $\{\{No,0\}, \{Si,100\}\}$ . Luego de esta manera se puede obtener el resultado de indicador parcial completitud funcional. De los indicadores parciales (con fondo gris en la tabla) *1.1 Completitud Funcional*, *1.2. Correctitud Funcional* y *1.3. Pertinencia Funcional*, se puede Obtener el valor del indicador global *1.Adecuación Funcional*.

Finalmente de los tres (3) indicadores globales *1.Adecuación Funcional*, *2. Usabilidad* y *3. Operabilidad* (estos últimos dos no están visibles en la Tabla 3) se puede obtener el valor final del EDI.

## Resultados

Luego de realizar el proceso de evaluación se obtuvieron los resultados finales para cada EDI, quedando así, en los tres (3) primeros lugares:

1. Arduino en La Escuela (AeLE) con un valor de 73,3225.
2. Facilino Offline (versión gratuita) con un valor de 72,3435.
3. Mixly con un valor de 70,9536.

Del análisis de los resultados surge:

Para la *1. Adecuación Funcional* AeLE fue el EDI que puntó más alto. A su vez, es el único que presenta mensajes y *warnings* cuando detecta que se producirá un error lógico, de compilación o de ejecución al posicionar los bloques. Mixly también presenta un buen número respecto a *1.2. Correctitud Funcional*, sin embargo, sufre de que ningún bloque tiene imágenes ilustrativas del dispositivo que controla, y también, presenta muchos errores de traducción.

Respecto a su *1.3.Pertenencia Funcional*, AeLE no destaca, incluso, hay un sensor utilizado en la secuencia que aún no está soportado en el EDI. Cabe aclarar que AeLE se encuentra en su fase beta y fue provista una copia del mismo para esta investigación.

En este apartado, es donde destaca Facilino, ya que es el único que cumple con todas las características, presentando todos los tipos de datos y sensores que se utilizan en la secuencia didáctica del TIP.

Ambos cuentan con una característica fundamental que es la de configurar los bloques que se decide tener en pantalla, permitiendo aplicar secuencias basadas en la didáctica por indagación en las que se presenta a los estudiantes un set de instrucciones más acotado para la solución de problemas. Esta característica también está presente en Facilino, sin embargo, no es tan flexible.

En cuanto a la característica de 2. *Usabilidad*, nuevamente destaca AeELE sobre los demás, teniendo una interfaz más accesible y personalizable. Todas sus categorías de bloques están distinguidas por colores e imágenes, incluso las subcategorías. Respecto a su capacidad de aprendizaje, para quienes han podido acceder a la versión beta, AeLE se ha distribuido con materiales tales como guías docentes y un taller en donde los estudiantes realizan proyectos enteros. Facilino cuenta en su página web con guías también destinadas a los estudiantes, pero poniendo el foco únicamente en aprender a utilizar sensores o actuadores específicos, y no en un proyecto entero como lo hacen los de AeLE.

Respecto a su 3. *Compatibilidad*, excepto Mixly, los otros dos EDI están disponibles para Windows, Linux y Mac. En el proceso de instalación y uso (Coexistencia), Mixly lleva la ventaja ya que es portable y no requiere la instalación de Arduino IDE, requisito indispensable para que el resto de los IDE puedan funcionar.

Para finalizar, desde esta investigación se recomienda a la cátedra IC el EDI AeLE, ya que es el que se destaca respecto a la correctitud funcional y la ayuda experta. Si bien Facilino cuenta con una mayor cantidad de bloques y sensores, su correctitud funcional es la más baja de los tres EDI.. Respecto a Mixly, es una opción intermedia de ambos EDI, tiene una gran cantidad de sensores utilizados en la secuencia y una buena correctitud funcional; sin embargo, está muy lejos de las características ofrecidas por AeLE.

## Conclusiones y trabajos futuros

En base a la necesidad de información detectada se continuó con el trabajo iniciado en 2021 sobre el que se realizó el proceso de medición aplicando las métricas diseñadas a los atributos del modelo de calidad.

Se diseñaron los indicadores elementales, globales y parciales con el objetivo de proporcionar una estimación de cada concepto calculable. Posteriormente, se llevó a cabo el proceso de evaluación que toma como insumo las medidas obtenidas en el proceso de medición y por medio de la aplicación de los indicadores las convierte en información contextualizada.

Luego del análisis de los resultados arrojados por el proceso de evaluación finalmente se alcanzó el objetivo que motivó la presente investigación: "recomendar el EDI que mejor se adapte a las necesidades del cuerpo docente del TIP".

Como trabajo futuro se plantea, por una parte, la incorporación de AeLE a la propuesta didáctica del TIP, y su prueba en la implementación del Taller en 2023. Por otra, extender/generalizar la presente

propuesta de medición para la evaluación de IDE para robótica por fuera de la propuesta didáctica del TIP.

## Referencias

Astudillo, G. J., Bast, S. G., & Willging, P. A. (2016). *Enfoque basado en gamificación para el aprendizaje de un lenguaje de programación*. *Virtualidad, Educación y Ciencia*, 7(12), 125-142.

Astudillo, G. J., Bast, S. G., Willging, P., Segovia, D., Castro, L., Lucero, P., Lobos, M., & Distel, J. M. (2019). *Estrategias innovadoras en los procesos de enseñanza y de aprendizaje de la programación*. XXI Workshop de Investigadores en Ciencias de la Computación, San Juan, Argentina. <http://sedici.unlp.edu.ar/handle/10915/77161>

Barrows, H. S. (1986). *A taxonomy of problem-based learning methods*. *Medical education*, 20(6), 481-486.

Covella, G. J. (2005). *Medición y evaluación de calidad en uso de aplicaciones web*. Universidad Nacional de La Plata. Recuperado de : <https://repo.unlpam.edu.ar/handle/unlpam/851>.

ISO/IEC (2011). *ISO/IEC 25010 - Systems and Software Engineering - Systems and Software Quality Requirements and Evaluation (SQuaRE) - System and Software Quality Models*. Technical report.

Kitchenham, B., & Pfleeger, S. L. (1996). *Software quality: the elusive target [special issues section]*. *IEEE software*, 13(1), 12-21.

Lobos, M., Bast, S., Astudillo, G. (2021). *Hacia una evaluación de calidad de Entornos de Desarrollo Integrado de Robótica Educativa*. *Actas JADiCC 2021*. pp. 78-85. Recuperado de: <https://jadicc.program.ar/wp-content/uploads/2022/03/actas-jadicc-2021.pdf>.

Monsalves González, S. (2011). *Estudio sobre la utilidad de la robótica educativa desde la perspectiva del docente*. *Revista de Pedagogía*, 32(90), 81-117.

Olsina, L., Papa, F., & Molina, H. (2008). *How to measure and evaluate web applications in a consistent way*. In *Web Engineering: Modelling and Implementing Web Applications* (pp. 385-420). Springer, London.

## **Taller de Programación de apps para celulares Generando vocaciones científicas en los jóvenes estudiantes de nivel secundario**

Antuña, Lucía / lucia.antuna15@gmail.com

Borbolla, Carolina / carob197@gmail.com

Cruz, Ivana / icruz@ing.unrc.edu.ar

Ferreira Szpiniak, Ariel / aferreira@exa.unrc.edu.ar

Universidad Nacional de Río Cuarto

### **Resumen**

El taller de "Taller de programación de apps con tu celular" se desarrolló en el año 2021 en el marco de la "Línea de Trabajo A: Visitas a escuelas para seguir difundiendo entre los adolescentes la importancia de aprender a programar" correspondiente a la Convocatoria 2019 de la Fundación Dr. Manuel Sadosky a presentar manifestaciones de interés para universidades nacionales con carreras informáticas en el marco de la iniciativa Program.AR. Posteriormente se presentó el taller "Programación de apps para celulares, generando vocaciones científicas en los jóvenes estudiantes de nivel secundario" como proyecto de extensión en la tercera convocatoria de la UNRC, fue aprobado y se está desarrollando en la actualidad.

El taller es desarrollado por la Universidad Nacional de Río Cuarto (UNRC), con la participación de la Facultad de Ciencias Exactas, Físico-Químicas y Naturales (Departamento de Computación), la Facultad de Ingeniería (Departamento de Telecomunicaciones), la Secretaría de Extensión y Desarrollo (Centro IRC), el Punto Digital Río Cuarto y Fundación por la Cultura de la Municipalidad de Río Cuarto. Durante el año 2021 se utilizó el material y metodología propuesta por la Fundación Dr. Manuel Sadosky. Fue llevado a cabo en el Gran Río Cuarto y la región, participaron 122 estudiantes de 82 escuelas, y se realizaron 38 talleres al cabo de 14 semanas. Actualmente, en el marco del proyecto de extensión, se utilizan los mismos materiales, pero con una metodología diferente, ya que se brindan talleres no presenciales y presenciales en el Punto Digital Río Cuarto.

**Palabras clave:** Programación, Celulares, Estudiantes, Secundario

### **1. Introducción**

El proyecto propone brindar un taller a los jóvenes para que aprendan a programar aplicaciones (apps) para teléfonos móviles, de una manera sencilla. El objetivo es introducir el concepto de pensamiento computacional, ayudar a descubrir vocaciones científicas entre los jóvenes e incentivar a los estudiantes secundarios para continuar sus estudios en carreras relacionadas con las Tecnologías de la Información y la Comunicación. La temática del taller está relacionada con la investigación realizada en el marco del proyecto "Pensamiento Computacional y los NAP de Educación Digital, Programación y Robótica desde un paradigma inclusivo y con compromiso social. En ese contexto se proponen los temas del taller, diseñando, construyendo, programando, probando y usando apps, de una forma lúdica y práctica.

Está destinado a estudiantes que cursan los últimos 3 años de escuelas secundarias (no excluyente) del Gran Río Cuarto y la región sur de la Provincia de Córdoba, que estén interesadas/os en conocer de qué se trata el mundo de la programación y usar su celular para crear pequeñas aplicaciones.

La UNRC tiene una larga trayectoria de articulación con escuelas de la ciudad y la región, con una política institucional de articulación y vinculación con los otros niveles del sistema educativo, que se ha venido desarrollando desde comienzos de la década de los noventa, basada en la convicción del derecho y el libre acceso a la educación universitaria pública y gratuita. Estas experiencias han tenido lugar en el marco de proyectos institucionales promovidos por las sucesivas gestiones académicas y también, en el marco de convocatorias ministeriales.

Desde el año 1992, la Facultad de Ciencias Exactas, Físico- Químicas y Naturales, de la cual

depende el Departamento de Computación, ofrece las carreras de Analista en Ciencias de la Computación, Licenciatura en Ciencias de la Computación y Profesorado en Ciencias de la Computación. Desde el año 1998, la Facultad de Ingeniería, de la cual depende el Departamento de Telecomunicaciones, ofrece la carrera de Ingeniería en Telecomunicaciones.

Ambas Facultades, a través de sus respectivos Departamentos, participan y organizan diversas actividades académicas, de extensión, pasantías, etc. en temáticas relacionadas con TIC en general, Ciencias de la Computación y Telecomunicaciones tendientes a fomentar su vinculación con el medio. También el Rectorado lleva adelante diferentes iniciativas para vincular a la Universidad con el sector productivo, científico, gubernamental, educativo y social. Las distintas acciones se desarrollan en el marco de Convenios con Ministerios Nacionales y Provinciales (como Educación y Ciencia y Tecnología), Agencias Provinciales (como Córdoba Innovar y Emprender), la Secretaría de Políticas Universitarias, la Mesa TIC del Gran Río Cuarto y convenios específicos con centros educativos de Río Cuarto y la región.

En tal sentido, y en relación específica con los talleres para estudiantes secundarios propuestos por la Iniciativa Program.AR, la UNRC participó desde la primera convocatoria, en 2016:

- **Convocatoria 2016:** se realizó el “**Taller de programación y animación 3D para escuelas secundarias del Gran Río Cuarto**”. Realizado mediante Protocolo de Trabajo con el Ministerio de Educación de la Provincia de Córdoba. La UNRC presentó una propuesta con la articulación de diferentes actores: Mesa TIC del Gran Río Cuarto, Facultad de Ingeniería, Facultad de Ciencias Exactas, Físico-Químicas y Naturales, y la Secretaría de Extensión y Desarrollo de la UNRC. Además, en el Centro Cívico Provincial, se realizó una jornada previa con Inspectores y Directivos de las escuelas beneficiarias, y se firmó un Protocolo de Trabajo con el Ministerio de Educación de Córdoba.

**Se realizaron 44 talleres, en 7 centros educativos del Gran Río Cuarto (Río Cuarto, Santa Catalina y Las Higueras), al cual asistieron 743 estudiantes.** Periodo: 2016-2017.

- **Convocatoria 2018:** se realizó el “**Taller de programación y animación 3D para escuelas secundarias de Río Cuarto y la región**”. La UNRC presentó una propuesta con la articulación de diferentes actores: Mesa TIC del Gran Río Cuarto, Facultad de Ingeniería, Facultad de Ciencias Exactas, Físico-Químicas y Naturales, y la Secretaría de Extensión y Desarrollo de la UNRC. Fue Declarado de Interés Educativo Provincial por el Ministerio de Educación de la Provincia de Córdoba (Resolución 867/18).

**Se realizaron 22 talleres, en 11 centros educativos del Gran Río Cuarto y la Región, al cual asistieron 435 estudiantes.** Periodo: 2018-2019.

Las escuelas participantes en la convocatoria 2016 y 2018 fueron las siguientes:

- IPET 258 - Mayor Ingeniero Francisco de Arteaga – CUE 140154300
- IPEM 27 - Dr. René Favalaro - CUE 140447500
- IPEM 28 - VILLA DE LA CONCEPCION DEL RIO CUARTO - CUE 140447600
- IPEM 95 - Mariquita Sánchez de Thompson - CUE 140299700
- IPET 26 - Juan Filloy - CUE 140447400
- IPEA 215 - Raúl Scalabrini Ortiz - CUE 140120300
- ProA Biotecnología - CUE 140566600
- CITER: RAMÓN ARTEMIO STAFFOLANI – CUE 140543100
- IPEM 95 Mariquita Sánchez de Thompson – PIT14/17 “San Pantaleón” - CUE 140299700
- IPET 79 Renato de Marco – CUE 140301300
- IPEM 27 René Favalaro Anexo La Carolina – CUE 140447502
- IPET 314 - Libertador General Don José De San Martín – CUE 140475700
- INSTITUTO SANTA ROSA DE LIMA – CUE 140482300
- IPEATyM 186 - Capitán Luis Darío José Castagnari – CUE 140153800

## 2. Caracterización de la situación problema y propuesta

A partir del Programa Conectar Igualdad, desde 2010 existen fuertes políticas públicas para reducir la brecha digital de 1° orden y 2° orden. Desde 2014, se viene dando un proceso para reducir la brecha digital de 3° orden, es decir, comprender cómo funcionan las computadoras y el mundo digital. En 2015, el Consejo Federal de Educación resolvió que la enseñanza de la programación es estratégica. A fines de 2018 se aprobaron los Núcleos de Aprendizaje Prioritarios de Educación Digital, Programación y Robótica (NAPEDPyR) que abarcan a todo el territorio nacional. Varios programas se han puesto en marcha, con menor o mayor impacto. Los NAPEDPyR buscan responder al imperativo de la época, la inclusión socio-educativa en medio de un veloz desarrollo científico-tecnológico atendiendo a las condiciones sociales en medio del territorio al cual se aplica. Los NAPEDPyR involucran estrictamente “una propuesta integral de innovación pedagógica y tecnológica que comprende como núcleos centrales el desarrollo de contenidos, el equipamiento tecnológico, la conectividad y la formación docente, que ayude tanto al desarrollo de las competencias de educación digital, como de las capacidades y saberes fundamentales” (Resolución CFE N.º 343/18). Uno de los aspectos sobresalientes de los NAPEDPyR es que las nuevas tecnologías sirvan para “(...) la integración plena a la sociedad y al mundo del trabajo” explicitando que una vez comprendido cómo se construyen los sistemas digitales y cómo se crean, puedan hacer uso crítico y creativo de las tecnologías (Resolución CFE N.º 343/18). en este sentido, el pensamiento computacional se impone como nueva forma de abordar y crear realidad y dada la existencia de una formulación adecuada de los NAPEDPyR. Por ese motivo se propone llevar adelante el proyecto denominado “Programación de apps para celulares. Generando vocaciones científicas en los jóvenes estudiantes de nivel secundario”, haciendo aportes significativos para la disminución de la brecha digital de 3° orden trabajando en el marco propuesto por los NAPEDPyR. La enseñanza del pensamiento computacional hoy en día se presenta como un desafío a alcanzar, debido a la caracterización que tiene esta ciencia en la sociedad. Uno de los primeros pasos a superar es desmitificar que es una ciencia difícil o inaccesible, sino todo lo contrario. Las nuevas metodologías de enseñanza que se utilizan para el aprendizaje de la computación apelan a entender cómo funciona la tecnología que rige el mundo actual, compartir saberes y gustos relacionados con la tecnología del software, la cultura colaborativa y descubrir vocaciones científicas. Como ya se ha evidenciado, el pensamiento computacional es una habilidad fundamental en el mundo en el que vivimos que nos permite independizarnos del rol de usuarios para ser creadores. Nos permite comprender más cabalmente la realidad que nos rodea, permitiéndonos actuar e incidir sobre la misma. Es una puerta de entrada muy amplia y no solo a las carreras informáticas que son una profesión con muy buena salida laboral.

La metodología de trabajo propuesta aborda el problema poniendo el foco en el estudiante. En los antecedentes presentados, se desarrolló dicha metodología y se lograron muy buenos resultados. La misma busca modificar la forma en que se enseña computación para desarrollar habilidades en los jóvenes como la abstracción y la resolución de problemas a través de proponer desafíos que pueden ser alcanzados por los estudiantes durante las clases planificadas. Para lograrlo se acompaña al estudiante durante todo el proceso formativo.

El objetivo principal del proyecto es brindar un taller a los jóvenes para que aprendan a programar aplicaciones (apps) para teléfonos móviles, de una manera sencilla. Entre los objetivos específicos podemos mencionar la introducción del concepto de pensamiento computacional; poner en práctica una propuesta integral de innovación pedagógica y tecnológica; ayudar a descubrir vocaciones científicas entre los jóvenes; incentivar a los estudiantes secundarios para continuar sus estudios en carreras relacionadas con las Tecnologías de la Información y la Comunicación; vincular la universidad con el nivel secundario del Gran Río Cuarto y la Zona Sur de la provincia de Córdoba; y dar continuidad a los talleres realizados en proyectos desarrollados anteriormente en la UNRC.

## 2.1 Metodología utilizada en el taller del año 2021

A continuación, se presenta el desarrollo realizado para la convocatoria 2019 del proyecto Línea de Trabajo A: Visitas a escuelas para seguir difundiendo entre los adolescentes la importancia de aprender a programar, en el marco de la iniciativa Program.AR, que se ejecutó durante la pandemia, en el año 2021.

El proyecto se enmarca dentro de las carreras de Profesorado en Ciencias de la Computación, Licenciatura en Ciencias de la Computación e Ingeniería en Telecomunicaciones. De la organización participaron la Facultad de Ciencias Exactas, Físico-Químicas y Naturales, la Facultad de Ingeniería y la Secretaría de Extensión y Desarrollo de la UNRC.

Las talleristas fueron tres estudiantes avanzadas del Profesorado en Ciencias de la Computación (dos) e Ingeniería en Telecomunicaciones (una), seleccionadas mediante una convocatoria abierta. Su actividad se enmarca dentro de las becas de vinculación que ofrece la UNRC y que son la herramienta utilizada en estos casos. La Secretaría de Extensión y Desarrollo se encargó de la gestión de dichas becas, así como también de todo lo concerniente al convenio y agenda entre la UNRC y la Fundación.

En el mes de marzo la Fundación Manuel Sadosky realizó la presentación formal del taller y brindó una capacitación virtual de 3 hs donde se presentó la metodología propuesta, contenidos sugeridos, etc. A partir de allí, el equipo de trabajo y las talleristas realizaron una serie de reuniones para definir la etapa de organización previa al dictado del taller, entre otras cosas se dispuso la utilización de una carpeta compartida en Google Drive, el armado de grupo de WhatsApp, etc. Como primera actividad se revisaron los materiales del taller (fichas de trabajo) y se adaptaron agregando contenido adicional como video-tutoriales y/o presentaciones multimedia. De este trabajo surgen los materiales del taller: Ficha del Proyecto 0, Ficha del Proyecto 1 y Ficha del Proyecto 2.

Para difundir la propuesta se creó un formulario de inscripción y una publicación con el enlace del mismo para realizar la convocatoria en diferentes medios como Instagram, WhatsApp, Facebook, Email, etc. Además, se confeccionó el texto del email de confirmación de la inscripción y otro para el envío de la información del grupo asignado. Las talleristas organizaron los grupos de cada taller incluyendo toda la información necesaria en una planilla alojada en la carpeta compartida. Allí se encuentran distribuidos los grupos en las diferentes semanas y horarios designados por cada una. Cada tallerista dictó un taller por semana.

A partir de la planilla de inscripciones, cada una de las talleristas procedió a la confirmación de la inscripción de los estudiantes asignados vía e-mail. Luego, se organizaron los grupos teniendo en cuenta la cantidad de estudiantes (5), los horarios de los estudiantes, escuela a la que pertenecen y curso. Una vez definidos los grupos, se procedió al envío de la información necesaria para realizar el taller: información general, metodología, materiales (enlace para descargar la app y fichas de trabajo), fecha y horarios asignados.

Para el dictado de cada taller, cada tallerista organizó un grupo de WhatsApp con los estudiantes a los fines de contar con un canal de comunicación rápido y allí hacer el seguimiento de la actividad previa al 1° encuentro virtual.

La metodología de dictado del taller fue 100% virtual, de forma asíncrona vía e-mail y WhatsApp y sincrónica mediante video-llamadas de Google Meet.

En el transcurso de 7 días se desarrollaron tres encuentros virtuales sincrónicos que contaban con la siguiente estructura:

- 1° encuentro: “Presentación del Proyecto 1”. Duración: 40 minutos
- 2° encuentro: “Presentación del Proyecto 2” Duración: 80 minutos
- 3° encuentro: “Cierre”. Duración: 40 minutos

Integrantes del equipo de trabajo

Coordinadores:

- Ariel Ferreira Szpiniak / Ivana Cruz / Pablo Etcheverry

Talleristas:

- Carolina Andrea Borbolla / Lucía Antuña / Lucrecia Mascanfroni

## 2.2 Metodología a utilizar en el taller del año 2022

A continuación, se presenta el desarrollo realizado para la Tercera Convocatoria a proyectos de extensión de la UNRC. Cabe destacar que el proyecto fue evaluado 9no, entre los 26 mejores (aprobados), sobre 68 presentados.

El proyecto se enmarca dentro de las carreras de Profesorado en Ciencias de la Computación, Licenciatura en Ciencias de la Computación e Ingeniería en Telecomunicaciones. De la organización participaron la Facultad de Ciencias Exactas, Físico-Químicas y Naturales, la Facultad de Ingeniería, la Secretaría de Extensión y Desarrollo de la UNRC (Centro IRC), el Punto Digital Río Cuarto y Fundación por la Cultura de la Municipalidad de Río Cuarto.

El proyecto propone continuar con la modalidad en formato taller, debido a los buenos resultados obtenidos en las experiencias anteriores. Las características del taller son las siguientes:

- Es gratuito y tiene una duración de 2 encuentros de 2 horas cada uno (4 horas en total).
- Es con modalidad presencial en el Punto Digital Río Cuarto y no presencial.
- Se replica en 10 oportunidades, cada 15 días, durante 5 meses.
- No requiere conocimientos previos.
- Cada estudiante trabaja desde su celular o computadora (propia o provista por la escuela).
- Los horarios se organizan de manera tal que no interfieran con las clases habituales en la escuela y que le quede cómodos a los jóvenes.
- Se proporciona todo el material necesario (vídeos explicativos y guías paso a paso en formato pdf) para el desarrollo del taller.
- Está a cargo de dos estudiantes universitarias (tutoras) de las carreras de Ciencias de la Computación y de Telecomunicaciones, con el acompañamiento de docentes de ambas carreras. Las estudiantes ya poseen experiencia dado que fueron quienes tuvieron a cargo los talleres de Sketchware (software para programar apps de celulares) durante 2021 en el marco de la convocatoria 2019 de la Iniciativa Program.AR.
- Es personalizado, no más de 10 estudiantes por taller, divididos en 2 grupos. Cada grupo tiene una tutora a cargo.

Integrantes del equipo de trabajo

- Directora: Ivana Cruz
- Co-Director: Ariel Ferreira Szpiniak
- Talleristas: Carolina Andrea Borbolla / Lucía Antuña
- Coordinador Punto Digital: Gastón Malgieri

## 2.3 ¿Qué es el Punto Digital?

Es un Programa impulsado por la Secretaría de Innovación Pública de la Nación cuyo objetivo es democratizar el acceso a las Tecnologías de la Información y la Comunicación (TIC), entendidas como un derecho fundamental en el desarrollo. Cuenta con acceso a Internet y una sala de aprendizaje con 16 computadoras, microcine con proyector y un espacio de entretenimiento. Hay 600 Puntos Digitales en todo el país. En Río Cuarto está a cargo de la Fundación por la Cultura de la Municipalidad de Río Cuarto.

## 2.4 Escuelas participantes en el año 2021

En total participaron estudiantes pertenecientes a 82 escuelas de las provincias de Córdoba, La Rioja, San Luis, Santa Fé, Buenos Aires y una escuela de Huancayo, Perú. De las cuales 77 escuelas



perteneían a la provincia de Córdoba y a las siguientes localidades: Río Cuarto, Las Acequias, Laboulaye, Achiras, La Calera, La Carlota, Sampacho, Alcira Gigena, Tosquita, Cruz del Eje, Villa de Soto, La Cumbre, Alejandro Roca, Gral Levalle, Vicuña Mackenna, Carnerillo, Gral Cabrera, Villa Reducción, Las Higueras, Mattaldi, Coronel Moldes, Huinca Renancó, ciudad de Córdoba, San Joaquín, Almafuerte, Serrezuela, Villa General Belgrano, Santa Isabel.

Las Escuelas participantes se resumen a continuación: Atica, C.E.N.M.A 24 Banda Norte anexo Las Acequias, CEMMA Antonio Sobral, CENMA 24, CENMA 73 Dr. Arturo Jauretche, Colegio Hispano Argentino, Colegio La Merced, Colegio Nuestra Señora del Carmen, Colegio San Juan de la Cruz, Colegio Santa Teresita, Cristo Rey, Domingo Faustino Sarmiento, EMPA 1234, Santa Isabel, EPET N° 1 "Gral Ángel V. Peñaloza", Establecimiento N 13 Comercial Ballester, Escuela Normal Juan Pascual Pringles, Escuela Normal Justo José de Urquiza, Escuela ProA (orientación: software), Escuelas Pías de Río Cuarto, Cba, Escuelas PRoA La Carlota con orientación en ciencias naturales, Galileo Galilei, Hogar escuela María Benita Arias, I.O.A.P.A.P Instituto Orientado y Agrotécnico Pablo A. Pizzurno, I.P.E.A N° 226 "Héroes de Malvinas", I.P.E.M 278 Malvinas Argentinas, I.P.E.M N°421 Tosquita Y Anexos Rurales Washington y Colonia La Argentina, I.P.E.M. 273 "Manuel Belgrano", I.P.E.T N°254, I.P.E.T. Nro. 259 "Ambrosio Olmos", I.P.P.C. Caviglia, Instituto 24 de Setiembre, Instituto José Manuel Estrada, Instituto La Consolata, Instituto Lanteriano Nuestra Señora de la Merced, Instituto Ntra. Sra. Del Carmen, Instituto Nuestra Madre de la Merced, Instituto Nuestra Señora de Lourdes, Instituto Privado Alejandro, Instituto Privado: Sagrada Familia, Instituto San Buenaventura, Instituto San Francisco de Asís, Instituto Santo Domingo, Instituto Secundario General Levalle, Instituto Secundario Juan Pascual Pringles, Instituto Técnico Adrián P. Urquía (ITAPU), Instituto Técnico Santa María Teresa Goretti, IPEA 291, IPEA N 127 Doctor Mateo Carusillo, IPEAyT N°186 Capitán Luis Darío José Castagnari, IPEM 128 Dr. Manuel Belgrano, IPEM 178 América Latina, IPEM 203 Dr. Juan Bautista Dichiará, IPEM 281 Dr. A. Carlos Lucero Kelly, IPEM 283 Fray Mamerto Esquiú, IPEM 94 Dolores Lavalle de Lavalle, IPEM 95 Mariquita Sánchez De Thompson, IPEM N° 141. Dr. Dalmacio Vélez Sarsfield, IPEM N° 195 Pedro Ignacio Bean, IPEM N° 90 Juana Manuela Gorriti, IPEM N°178 "América Latina" Anexo San Joaquín, IPEM N°28 Villa de la Concepción, IPEM N°299 Dr. Spiridon Estéfano Naumchik, IPEM N°27, IPEM N°29 Felipe Galizia, IPEMyT 280 Agustín Tosco, IPET N°106 Remedios de escalada, IPET 254 Tristán de Tejeda, IPET N°76 Gustavo Riemann, IPET N°258 "FRANCISCO DE ARTEAGA", IPET N°26 Juan Filloy, IPETYM N°257 "Dr. René Favalaro" Laboulaye, La Merced, Manuel Belgrano, Mariano Moreno, PROA Desarrollo de software, PROA sede La Carlota, PROA-Biotecnología, SALESIANO, San Francisco de Asís, Santa Eufrasia.

## **2.5 Organización de los estudiantes que participaron en el año 2021**

A medida que las/os estudiantes realizaban la inscripción se fue realizando el armado de los grupos. Cada grupo se formó en función de parámetros tales como horarios y cursos a los que pertenecían. Para la organización de los grupos se construyó una tabla que identificaba el número de grupo, apellido, nombre, DNI, correo electrónico y curso de las/os estudiantes. Se conformaron una totalidad de 38 grupos. Las/os estudiantes que participaron de los talleres se encontraban cursando los años 1er, 2do, 3ro, 4to, 5to, 6to o 7mo.

## **2.6 Evidencias del trabajo realizado por los alumnos durante el taller del año 2021**

A continuación, se pueden observar algunos de los trabajos realizados por los estudiantes a lo largo del taller. El taller consistió en la realización de actividades denominadas Proyectos. La imagen 1 corresponde al "Proyecto 0", las imágenes 2, 3, 4 y 5 corresponden al Proyecto 1" y las imágenes 6 y 7 corresponden al "Proyecto 2". Se puede notar que, en cada trabajo realizado, las/los estudiantes emplearon su creatividad, tanto en el diseño de las aplicaciones como así también en la programación de las mismas.



Imagen 1: Proyecto 0

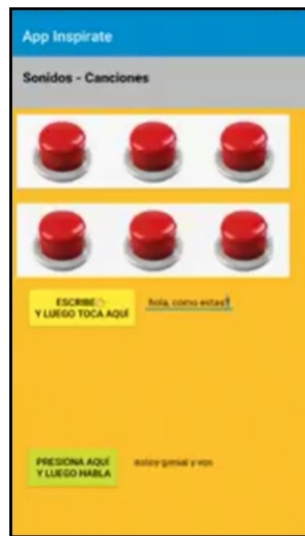


Imagen 2: Proyecto 1

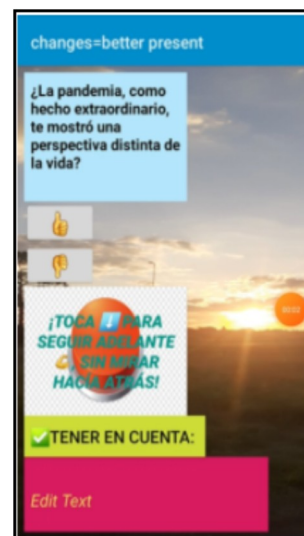


Imagen 3: Proyecto 1



Imagen 4: Proyecto 1

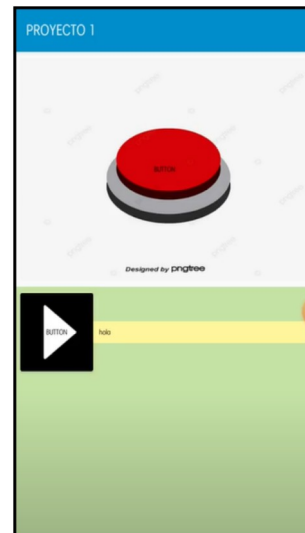


Imagen 5: Proyecto 1

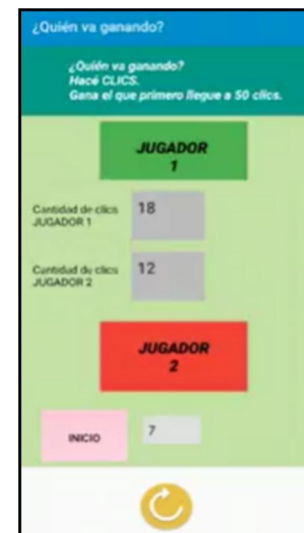


Imagen 6: Proyecto 2

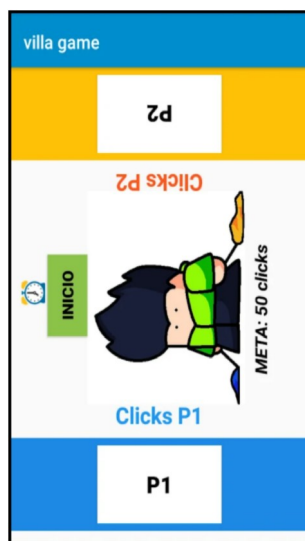


Imagen 7: Proyecto 2

### 3. Resultados de la encuesta realizada a los alumnos durante el taller del año 2021

Se organizó una encuesta para medir el desempeño del taller. La misma fue compartida a los 122 estudiantes participantes por email y WhatsApp. En la Imagen 8 se observa la respuesta a la pregunta: *¿Crees que el taller ayudó a conocer de que se trata el mundo de la programación?* En la Imagen 9 se observa la respuesta a la pregunta: *¿Qué tan interesante fue la propuesta del taller de programación de apps con tu celular?* En la Imagen 10 se observa la respuesta a la pregunta: *¿Cómo te resultaron los desafíos propuestos en el taller?* En la Imagen 11 se observa la respuesta a la pregunta: *¿Estás satisfecho con tu trabajo a lo largo del taller? ¿Cumplió tus expectativas?* En la Imagen 12 se observa la respuesta a la pregunta: *¿Recomendarías el taller a compañeras/os de tu escuela?* En la Imagen 13 se observa la respuesta al planteo: *A partir de la participación en este taller, ¿pudiste visualizarte en el futuro como estudiante de carreras relacionadas con las tecnologías de la información y la comunicación (TIC)?* En la Imagen 14 se observa la respuesta al planteo: *Nuestra universidad cuenta con una amplia propuesta de cursos auto asistidos relacionados con esta temática. ¿Estarías dispuesto/a participar de alguno de ellos en el corto plazo?* En la Imagen 15 se observa la respuesta a la pregunta: *¿Qué te pareció la convocatoria al taller, inscripción, contacto de las docentes, explicaciones, organización de los grupos, metodología, materiales y modalidad?* En total se recopilaron 58 respuestas de los estudiantes participantes, a continuación, se puede observar el resultado de las mismas sistematizado en gráficos circulares.

¿Creés que este taller ayudó a conocer de qué se trata el mundo de la programación?  
58 respuestas



Imagen 8

¿Qué tan interesante fue la propuesta del taller de programación de apps con tu celular?  
58 respuestas

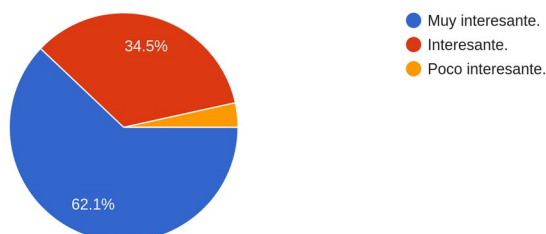


Imagen 9

¿Cómo te resultaron los desafíos propuestos en el taller?  
58 respuestas



Imagen 10

¿Estás satisfecho con tu trabajo a lo largo del taller? ¿Cumplió tus expectativas?  
58 respuestas



Imagen 11

¿Recomendarías el taller a compañeras/os de tu escuela?  
58 respuestas

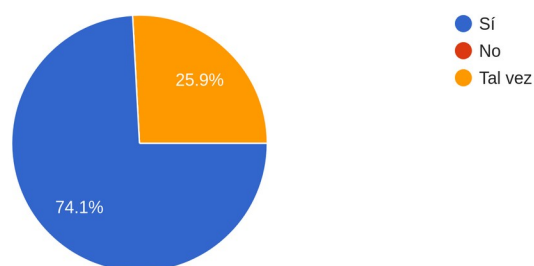


Imagen 12

A partir de la participación en este taller, ¿pudiste visualizarte en el futuro como estudiante de carreras relacionadas con las tecnologías de la información y comunicación (TIC)?

58 respuestas

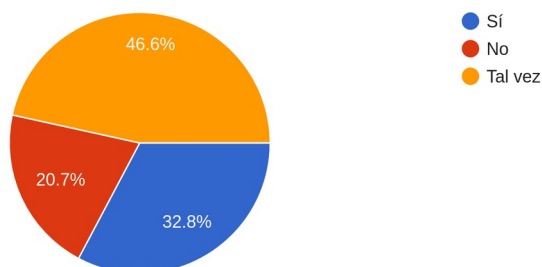


Imagen 13

Nuestra universidad cuenta con una amplia propuesta de cursos autoasistidos relacionados con esta temática. ¿Estarías dispuesto/a a participar de alguno de ellos en el corto plazo?

58 respuestas

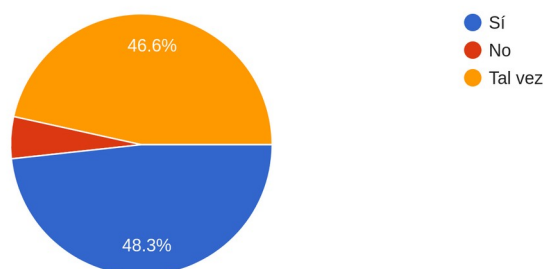


Imagen 14

¿Qué te pareció la convocatoria al taller, inscripción, contacto de las docentes, explicaciones, organización de los grupos, metodología, materiales y modalidad?

58 respuestas

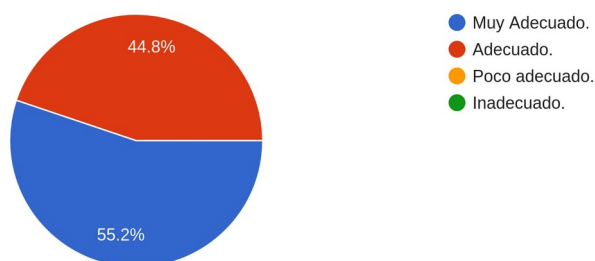


Imagen 15

Además, en el formulario se ofrecía un cuadro para responder a la siguiente pregunta:

¿Hay algo más que te gustaría compartir con nosotros sobre el taller o la universidad? De las 58 respuestas se pueden resaltar las siguientes contestaciones:

- Que siempre hagan estos tipos de talleres, para que muchos estudiantes puedan realizar

actividades que les interese o que siempre han querido hacer, también para estudiantes que no sepan mucho del tema y puedan adentrarse en este mundo y en el Mundo de la informática.

- Es una linda experiencia que lleva a tener un poco más de conocimiento en algo que vemos todos los días.
- Fue un taller interesante en donde la pasé bien y adquirí nuevos conocimientos gracias a éste.
- Fue interesante tener la oportunidad de participar, poder aprender y ver que conflictos tiene el otro que a lo mejor a nosotros fue algo que nos resultó más fácil y lo hicimos con mayor agilidad, es decir, desafiar nuestras capacidades.
- Fue una buena propuesta me gustó mucho
- Buena organización, profesores atentos e interesados en las/os estudiantes
- ¡¡Me encantó esta oportunidad!!, muchas gracias por ofrecer ampliar conocimientos!!
- No tengo críticas negativas, estoy muy contenta con el taller y espero pronto conocer la universidad.
- La propuesta me pareció muy útil ya que me dio una base de lo que es programar y me parece que es muy útil para ayudar a aquellas personas que aún buscan un futuro para ellos.
- Todo muy bueno.... me pude decidir qué estudiar
- ¡¡Me gustaría compartir que este taller me gustó mucho más que todo creando los juegos, Lo recomendaría mucho!!

#### **4. Conclusiones y síntesis en números**

En el año 2021, el proyecto fue llevado a cabo en el Gran Río Cuarto y la región, donde 211 estudiantes realizaron la inscripción, de 82 escuelas, de los cuales 122 estudiantes participaron efectivamente del taller. Se conformaron y dictaron 38 talleres al cabo de 14 semanas.

Como resultado de la encuesta, donde el 47,6% de las/os estudiantes participantes dieron respuesta a la misma, el 96,6% afirmaron que les resultó interesante el taller, como así también la gestión y materiales brindados les resultaron adecuados. Además, el 74,1% afirmó que recomienda el taller y el 25,9% tal vez lo recomendaría. Esto brinda un resultado prometedor para la continuidad del proyecto.

Un 91,4% de los estudiantes participantes creen que les ayudó a conocer el mundo de la programación, el 89,7% se manifestó satisfecho con su trabajo en el taller. Además, el 63,8% indicaron que las actividades resultaron desafiantes y al 27,6% les resultaron simples. Incluso el 32,8% asienta que se visualiza como un futuro estudiante de carreras relacionadas con las nuevas tecnologías de la información y comunicación, un 46,6% indica que tal vez y un 20,7% afirma que no. Por último, en referencia a la pregunta abierta de la encuesta, los estudiantes en su mayoría se consolidaron satisfechos con el taller y, además, esperan ser partícipes de más talleres o cursos que alimenten su conocimiento.

En el año 2022, los talleres que se desarrollan en el marco de los proyectos de extensión de la Secretaría de Extensión y Desarrollo de la Universidad Nacional de Río Cuarto, a través del proyecto denominado “Programación de apps para celulares, generando vocaciones científicas en los jóvenes estudiantes de nivel secundario”. Para su desarrollo se articuló con las Inspecciones de la Zona XII de la Dirección General de Educación Técnica y Formación Profesional (DGETyP) y de la Regional Pte. Roque Sáenz Peña/Gral Roca de la Dirección General de Educación Secundaria (DGESEC).

El proyecto que se lleva adelante durante todo el año 2022 buscar dar continuidad a las actividades desarrolladas en años anteriores. Los resultados obtenidos permiten utilizar la metodología que maximiza el objetivo, el formato taller y la articulación con estudiantes universitarios es de gran importancia para acompañar a los jóvenes a descubrir vocaciones científicas e incentivar a los estudiantes secundarios para continuar sus estudios en carreras relacionadas con las Tecnologías de la Información y la Comunicación u otras carreras que se ofrecen en la UNRC. Considerando el resultado de la encuesta realizada a los estudiantes que completaron el taller de programación de

Apps en 2021 se planificó reforzar, durante el taller, la difusión de las carrera Técnico/Informáticas, explicando no solo la ambientación a la vida universitaria que requieren los estudiantes en su paso por la universidad sino también explicar con ejemplos concretos los puestos de trabajo orientados a 4 perfiles, emprendedor, trabajo en el medio productivo, en docencia y el perfil de trabajo en investigación.

En el mes de abril se realizó una convocatoria y un período de inscripción durante 15 días para cubrir los talleres a desarrollar en el primer semestre del año. El resultado arrojó 144 estudiantes inscriptos. En el formulario de inscripción se le solicitó a las/os estudiantes que eligieran la modalidad de dictado de los talleres. La mayoría eligió el formato virtual porque son de escuelas de la región.

Para el desarrollo del taller se conformaron en 9 grupos de 11 estudiantes cada uno, para el taller en formato virtual, y 3 grupos de 15 estudiantes para el taller en formato presencial. Así mismo, se realizó un convenio con el Punto Digital Río Cuarto que posibilita desarrollar los talleres presenciales en un lugar cercano al centro de la ciudad de Río Cuarto (el campus universitario está en las afueras). Durante el segundo semestre se continuará con el desarrollo de los talleres, además se abrirá una nueva convocatoria para cubrir el mismo con más estudiantes interesados y se visitan escuelas que han manifestado el interés por realizar el taller en sus propias instalaciones. Dado el éxito de la propuesta, se planea continuar con esta actividad en el futuro.

## 5. Bibliografía y referencias

- Ferreira Szpiniak, A; Etcheverry, P. (2019). *Herramientas para descubrir vocaciones científicas y vincular la universidad con la escuela secundaria: talleres, olimpiadas y festivales de robótica y programación*. Segundas JADiPro. Córdoba. 7 y 8 de junio 2019.
- Wing, J. M. (2011). *Computational Thinking: What and Why?* The Link Magazine, Spring.
- *Iniciativa Program.AR*. Fundación Sadosky. <https://program.ar/>
- *Informe final del "Taller de programación de apps con tu celular" - UNRC* (2021). Línea de Trabajo A: Visitas a escuelas para seguir difundiendo entre los adolescentes la importancia de aprender a programar. Convocatoria 2019, Fundación Dr. Manuel Sadosky. Program.AR.
- *Núcleo de Aprendizajes Prioritarios de Educación Digital, Programación y Robótica*. Educ.ar (2018). Recuperado el 20 de junio de 2022 de: <https://www.educ.ar/recursos/150123/nap-de-educacion-digital-programacion-y-robotica?from=86>
- *Orientaciones pedagógicas de Educación Digital*. Ministerio de Educación de la Nación (2017). Recuperado el 20 de junio de 2022 de: <https://www.educ.ar/recursos/132262/orientaciones-pedagogicas-de-educacion-digital>
- *Programación y robótica: objetivos de aprendizaje para la educación obligatoria*. Ministerio de Educación, Cultura, Ciencia y Tecnología de la Nación (2017). Recuperado el 20 de junio de 2022 de: <https://www.educ.ar/recursos/132339/programacion-y-robotica-objetivos-de-aprendizaje-para-la-educacion-obligatoria>
- *Resolución CFCyE N° 225/04* (2004). Recuperado el 20 de junio de 2022 de: <https://cfe.educacion.gob.ar/resoluciones/res04/225-04.pdf>
- *Resolución Consejo Federal de Educación N° 263/15* (2015). Recuperado el 20 de junio de 2022 de: <https://cfe.educacion.gob.ar/resoluciones/res15/263-15.pdf>
- *Resolución Consejo Federal de Educación N° 343/18* (2018). Recuperado el 20 de junio de 2022 de: [https://www.argentina.gob.ar/sites/default/files/res\\_cfe\\_343\\_18\\_0.pdf](https://www.argentina.gob.ar/sites/default/files/res_cfe_343_18_0.pdf)



## “SOMOS MUY POCAS MUJERES”.

### Diferencias de Género en carreras de Computación en los casos de UNC y UNComa y la búsqueda de referentes mujeres en el área.

Echeveste, M Emilia<sup>1</sup>; Rodriguez, Jorge<sup>2</sup>; Gomez, Marcos<sup>3</sup>;  
Moran, Marina<sup>4</sup> y Benotti, Luciana<sup>5\*</sup>

<sup>1,3,5</sup>Facultad de Matemática, Astronomía, Física y Computación- (FaMAF-UNC)

<sup>2,4</sup>Facultad de Informática - (UNComa)

<sup>1</sup>[emilia.echeveste@unc.edu.ar](mailto:emilia.echeveste@unc.edu.ar) <sup>2</sup>[j.rodrig@fi.uncoma.edu.ar](mailto:j.rodrig@fi.uncoma.edu.ar)

<sup>3</sup>[marcos.gomez@unc.edu.ar](mailto:marcos.gomez@unc.edu.ar) <sup>4</sup>[marina@fi.uncoma.edu.ar](mailto:marina@fi.uncoma.edu.ar) <sup>5</sup>[luciana.benotti@unc.edu.ar](mailto:luciana.benotti@unc.edu.ar)

#### Resumen

En este escrito analizamos las estadísticas realizadas por la Universidad Nacional de Córdoba (UNC) y la Universidad de Comahue (UNComa) acerca de la composición de la matrícula según el género en los ingresos y egresos de las carreras de Informática y Computación entre 2016 y 2020. Los datos muestran que la representación femenina en la población estudiantil ingresante resulta no equitativa, no superando el 20% en las carreras de grado y el 25% en las tecnicaturas. En relación a los egresos, los porcentajes no sólo preocupan con respecto a la perspectiva de género sino también en cuanto al decaimiento del número de estudiantes que logran finalizar sus estudios superiores. Para complementar estos datos, se realizaron encuestas a estudiantes de materias de primer año de las carreras analizadas en las estadísticas, con el objetivo de profundizar en las variables de género que permitan reflexionar sobre la baja matrícula femenina en estas carreras informáticas. En el análisis de estas respuestas vinculadas a los intereses por la disciplina aparecen en su mayoría referentes masculinos y escasa mujeres ligadas al oficio de la computación.

**Palabras clave:** Género, Equidad, Educación Universitaria, Ciencias de la Computación

#### Introducción

En 2017 la Comisión de la Condición Jurídica y Social de la Mujer de las Naciones Unidas (CSW 61) pronunció la importancia de la participación de las mujeres en la nueva fuerza de trabajo digital, al igual que la UNESCO (2018) reconoció lo esencial de tener en cuenta la dimensión de género en el progreso tecnológico, en donde los avances podrían verse distorsionados por sesgos y prejuicios (UNESCO, 2018). La Agenda 2030 de Desarrollo Sostenible, pone énfasis en considerar a las niñas y las mujeres como factores e impulsoras del cambio y no meras beneficiarias.

Por otro lado, el número de mujeres en el campo del desarrollo de software, el liderazgo tecnológico y otros tipos de roles capaces de influir de manera significativa sobre la innovación futura es aún menor. Por esto, es preciso alentar a las mujeres a participar en los campos STEM como creadoras, emprendedoras, innovadoras y líderes, además de eliminar los obstáculos que todavía enfrentan (CSW 61, 2017).

Entre 2015 y 2017 el número de trabajadores con habilidades en Inteligencia Artificial aumentó en un 190% a nivel global (Foro Económico Mundial -WEF-, 2018). Según datos compilados por LinkedIn y publicados en el Informe Global sobre la Brecha de Género del



Foro Económico Mundial, solo el 22% del total de profesionales que trabajan en el campo de la inteligencia artificial en todo el mundo son mujeres (WEF, 2018).

A nivel de visibilidad y divulgación social se han realizado esfuerzos importantes para promover la igualdad de género en STEM, por ejemplo a través de conmemoraciones que otorgan visibilidad en la comunidad. Una de ellas es la declaración del *Día Internacional de la Mujer y la Niña en la Ciencia* por parte de la Asamblea General de la ONU (A/RES/70/212) en el año 2015, que se celebra el 11 de febrero para reconocer el papel clave que poseen las mujeres y las niñas en las comunidades científicas y tecnológicas. O el cuarto jueves de abril se celebra el *Día Internacional de las Niñas en las TIC* buscando alentar y empoderar a las niñas y a las jóvenes para que consideren la posibilidad de emprender estudios y carreras profesionales en el campo de las TIC.

Como muestran los datos presentados anteriormente, es constante la demanda laboral en sectores vinculados al trabajo en tecnologías digitales. Sin embargo, las carreras que forman recursos humanos para ocupar esos puestos también presentan un bajo número en las matrículas, porcentaje que disminuye aún más cuando se trata de mujeres e incluso de aquellas personas que no se reconocen en el binarismo de hombre-mujer. En este escrito reconocemos las disidencias de género pero no hacemos una profundización en ello. Es de nuestro interés y quedará como pendiente incorporar en próximas producciones. Consideramos que la masculinidad es un modelo social que impone un modo particular de configuración de la subjetividad, la corporalidad y la posición existencial del común de los varones y que se traduce en el general de la sociedad a través de modelos androcéntricos y patriarcales que nos van atravesando. Este modelo aísla también otras formas de ejercer la masculinidad que se consideran disidentes (no heterocentradas). Existen relaciones de dominación no solo de los varones hacia las mujeres, sino entre los propios varones, en donde no es lo mismo ser varón heterosexual que homosexual, blanco que de color. Todas estas intersecciones producen diferentes procesos de subalternización. (Wigdor, 2016).

Argentina reportaba en 2010 sólo un 18 % de mujeres en el estudiantado en informática y los datos analizados por nuestro equipo muestran que actualmente la matrícula femenina en carreras de computación no aumentó proporcionalmente en los últimos 10 años aunque sí aumentó la matrícula global (Echeveste et al, 2021). Esta situación es similar en países limítrofes como Uruguay, en donde reportes de 2011 registraron que los ingresos de estudiantes mujeres a la educación universitaria superan al de los varones a nivel de grado; sin embargo es comparativamente minoritaria la proporción de las mujeres que optan por las ingenierías. Puntualmente en las Universidad de la República las preferencias de las mujeres se encontraban sesgadas hacia las ciencias sociales y humanitarias, medicina y ciencias de la salud, reproduciendo estereotipos de carreras feminizadas. Según datos de investigaciones chilenas, se registraron en el 2018, que sólo una de cada cuatro matriculadas en las áreas STEM es mujer y sólo un 5% de las mujeres trabajan en áreas tecnológicas (Soria-Barreto y Zuniga-Jara, 2020; Pineda y González, 2016).

Naciones Unidas ha solicitado, a quienes se vinculen en espacios responsables de la educación, introducir medidas dirigidas a la reducción de las barreras que limitan la participación de las mujeres en STEM. En esta línea, hemos revisado datos de género encontrados en dos materias del primer año de estudios de dos universidades nacionales para comenzar a indagar qué sucede en nuestras facultades y así poder pensar derivaciones en las

prácticas docentes e institucionales que habiliten una mayor circulación de las mujeres y disidencias en carreras de computación. Se trabajó con encuestas en la materia “Introducción a los Algoritmos”, una de las primeras asignaturas de la carrera de Ciencias de la Computación de la Universidad Nacional de Córdoba (UNC) y la asignatura “Introducción a la Computación”, materia del primer cuatrimestre, también del primer año de Licenciatura en Ciencias de la Computación (LCC), Licenciatura en Sistemas de Información (LSI) y Profesorado en Informática (PI) dictadas por la Universidad de Comahue (UNComa). A continuación, enumeramos las secciones de nuestro escrito que intentan seguir aportando a los debates teóricos sobre los géneros en las carreras tecnológicas. No pretendemos arribar a verdades cerradas sino seguir construyendo interrogantes que nos permitan pensar en la complejidad de la problemática.

El resto de este trabajo está organizado como sigue. En la próxima sección se analiza la participación de las mujeres en la computación, enfocando la atención sobre nuestras universidades y en particular sobre las carreras computacionales que se dictan en la Universidad Nacional de Córdoba y la Universidad Nacional del Comahue. La Sección Referentes en el área de las Ciencias de la Computación, explora las opiniones de la población consultada acerca de cuáles personas y sucesos resultaron relevantes en la definición de su proyecto de vida profesional. Mientras que en la sección que sigue, se analiza la percepción estudiantil en relación a la desigualdad de género en el dictado de las materias y en las carreras. Finalmente, el texto cierra poniendo en discusión algunas reflexiones elaboradas en el contexto de este trabajo.

## **2- ¿Existen carreras para mujeres y carreras para hombres?**

Las dicotomía entre lo masculino y lo femenino también se representan en las distintas profesiones, en donde históricamente algunas carreras se relacionaban a las mujeres y otras a los hombres. Disciplinas vinculadas a las llamadas ciencias “duras”, como Ingenierías e incluso las Ciencias de Computación, mantienen un reducto mayoritariamente masculino, mientras que aquellas vinculadas al cuidado y las humanidades muestran mayor representación femenina. Diferenciación que reconoce un proceso de disciplinamiento que encauzan como normales y “naturales” procesos que soslayan discriminación de género (Foures y Aguirre, 2020). La investigación realizada en la Universidad de Rio Negro por Foures y Aguirre (2020) analiza la presencia de mujeres en carreras masculinizadas, en donde si bien, se lograron avance en normativas que regulan, como por ejemplo, los Protocolos para la Actuación en Situaciones de Violencias Directas e Indirectas basadas en el Género, no resuelven por completo procesos cotidianos que podrían estar reproduciendo mecanismos de discriminación por género, consolidados históricamente desde un sistema patriarcal. Sin embargo, se reconocen movimientos con fuerzas instituyentes, en cuestionamientos y críticas a estereotipos sociales y culturales propios de un sistema androcentrista resquebrajado en su potencia de única verdad posible.

Investigaciones en el área de género y tecnología presentaron algunos indicadores que generan desventajas en infantes, en cuanto a sus expectativas futuras de elegir carreras

relacionadas con las TIC o ingenierías, que según datos PISA representan un 2 % y 22 % de las mujeres, respectivamente (PISA, 2015; PISA, 2017). Estos indicadores se relacionan con obstáculos de tipo socioeconómico, ausencia de modelos de mujeres, presencia de representaciones sociales estereotipadas y baja autopercepción de las niñas en cuanto a autoeficacia de sus habilidades en telecomunicaciones e informática (Cheryan, 2012; ICILS, 2013; UNESCO, 2017).

## **2-a ¿Qué pasó con las mujeres en la Computación?**

En el campo de la computación, hasta los años 80 la participación y el interés de la mujer en este campo presentaba porcentajes y crecimientos paralelos al resto de las áreas, y al de los hombres. Si bien no hay una respuesta clara y única sobre la caída del interés de las mujeres en este ámbito, una de las hipótesis se vincularía con la aparición del ordenador personal y la campaña publicitaria para introducirlo en los hogares, orientada exclusivamente al público masculino (Henn, 2014). Esto nos permite mirar la divulgación y las y los referentes que hay en el área, lo cual nos evidencia la posibilidad de establecer vínculos positivos o negativos con una disciplina y su posterior elección o no, de formalizarla en estudios formales y superiores.

Investigaciones argentinas y uruguayas (Berro, 2021 y Echeveste et al, 2021) mencionan registrar como importante la relación de género en las primeras etapas de la vida, en dónde se suele definir a la computación como un territorio masculino. Inti Berro (2021) de la Universidad de la República menciona en su tesis seis factores que desalientan la participación de las mujeres en las carreras relacionadas con la computación: 1) Estereotipos de género generados fuertemente en las primeras experiencias en edades más tempranas, 2) La socialización y los contextos, en dónde aparecen las expectativas de la familia y de los grupos de pares. 3) Acoso y discriminación de género, que se vinculan a establecer diferencias en niveles de inteligencia por ser mujer, incluso se mencionan estrategias que realizan las mujeres ante la necesidad de adaptarse a un medio masculinizado, cómo por ejemplo tener comportamientos específicos que les permitan convivir en ese espacio o una vestimenta similar a la de sus colegas varones para así tener menos exposición. 4) Factores del ámbito laboral tanto el sector privado como académico, que presenta una “segregación vertical”, dificultad para acceder a determinados puestos, y una “segregación horizontal” vinculada a la distribución por género en las tareas al interior del sector tic. Para atender a este factor, la autora propone empezar a considerar “currículum ciegos” para evitar los estereotipos de género que afectan las oportunidades laborales de las mujeres ya que se considera que no evalúan los currículum de varones y mujeres al mismo grado. 5) El sistema educativo aparece con otro factor en el que se emplean prejuicios, por ejemplo, de docentes en relación a las disciplinas y prácticas de mujeres y de varones. Por último, 6) la dicotomía trabajo-familia, en especial vinculados por el deseo de ser madre, las cuales se ven sobrecargadas en los que respecta las tareas del hogar y de cuidado, problemática que se observó en aumento en la pandemia.

El hecho de que actualmente las Ciencias de la Computación, como también las carreras profesionales ligadas a éstas, se encuentran con estructuras masculinizadas, ha motivado no sólo la consideración única de modelos masculinos en los desarrollos e investigaciones (visión androcéntrica), sino también la pérdida de referencias de mujeres

que han sido y son importantes en la disciplina y que su mayor visibilización actuaría como modelos de referencia para muchas niñas y mujeres (Moreda, 2021). Si pensamos en estas prácticas a un nivel micro, el trabajo de género impulsado por el Grupo de Trabajo de Igualdad de Género de la Red Vives de universidades de España, menciona que la reducida participación de las mujeres en el sector de la tecnología “motiva la presencia hegemónica de valores androcéntricos y sexistas, tanto en el conocimiento que se transmite en las aulas como en los productos y tecnologías informáticas que hoy podemos encontrar en el mercado”. (Moreda, 2021, p. 7)

En el siguiente apartado desarrollaremos los análisis que realizamos en torno a los ingresos y egresos de las carreras de informática de dos universidades nacionales, la Universidad Nacional de Córdoba y la Universidad Nacional del Comahue.

## 2.b ¿Qué pasa en nuestras universidades?

Como menciona el reporte de la Red Vives de Universidades, es importante destacar que “*los entornos con sobrerrepresentación masculina son espacios disuasorios para la participación de las mujeres*” (Moreda, 2021, 13). En este sentido, revisamos los anuarios de 2016 a 2020<sup>1</sup> del Programa de Estadísticas Universitarias de la Universidad Nacional de Córdoba (UNC) y un reporte elaborado por la Secretaría Académica de la Universidad de Comahue (UNComa) acerca de la composición de la matrícula en la Facultad de Informática<sup>2</sup>.

La información recuperada nos presenta datos sobre los ingresos y egresos según género femenino y masculino en carreras informáticas dictadas en la Universidad Nacional de Córdoba y la Universidad Nacional del Comahue. Se trabajó sobre las carreras Ingeniería en Computación (Facultad de Ciencias Exactas, Físicas y Naturales - UNC) y Licenciatura en Ciencias de la Computación (FaMAF-UNC) dictadas en la ciudad de Córdoba y sobre las carreras Licenciatura en Ciencias de la Computación, Licenciatura en Sistemas de Información, Profesorado en Informática, Tecnicatura Universitaria en Administración de Sistemas y Software Libre y Tecnicatura Universitaria en Desarrollo Web dictadas en la ciudad de Neuquén por la Facultad de Informática - UNComa. No forman parte de este estudio las titulaciones intermedias para las licenciaturas dictadas en ambas universidades.

Las Tabla 1 y Tabla 2 aportan información acerca de la composición de la población estudiantil ingresante a las carreras relacionadas a computación dictadas en la Universidad Nacional de Córdoba y la Universidad Nacional del Comahue. Los datos muestran que la representación femenina en la población estudiantil ingresante resulta no equitativa, no superando el 20% en las carreras de grado y el 25% en las tecnicaturas.

<sup>1</sup> Se seleccionaron estos años ya que son últimos 5 anuarios estadísticos de UNC publicados <https://www.unc.edu.ar/programa-de-estad%C3%ADsticas-universitarias/anuarios-estad%C3%ADsticos>

<sup>2</sup> Se trabajó sobre un reporte, elaborado por la Secretaría Académica, destinado específicamente a este estudio.

Tabla 1. Cantidad de estudiantes que ingresaron en carreras de computación de la UNC (2016-2020)

<b>CANTIDAD DE INGRESOS UNC (2016-2020)</b>						
<b>Año</b>	<b>Ingeniería en Computación</b>			<b>Licenciatura en Ciencias de la Computación</b>		
	Mujeres	Hombres	Total	Mujeres	Hombres	Total
<b>2020</b>	48 (12%)	341(88%)	389	125 (17%)	611 (83%)	736
<b>2019</b>	28 (13%)	189 (87%)	217	61 (18%)	294 (82%)	355
<b>2018</b>	22 (12%)	161(88%)	183	39 (15%)	222 (85%)	261
<b>2017</b>	29 (15,5%)	159 (84,5%)	188	28 (17%)	141 (83%)	169
<b>2016</b>	16 (12%)	116 (88%)	132	22 (18%)	103 (82%)	125

Estos datos vuelven a demostrar que si bien la matrícula aumenta a medida que pasan los años, incluso de 2019 a 2020 se produce un aumento de más del doble de estudiantes en la licenciatura y más de la mitad en la carrera de ingeniería. Sin embargo, el porcentaje no se modifica con los años y continúa siendo significativamente menor en cuanto a la matrícula femenina.

Tabla 2. Cantidad de estudiantes que ingresaron en carreras de computación de la UNComa (2016-2020)

<b>CANTIDAD DE INGRESOS UNComa (2016-2020)</b>						
<b>Año</b>	<b>Carreras de Grado</b>			<b>Tecnicaturas</b>		
	Mujeres	Hombres	Total	Mujeres	Hombres	Total
<b>2020</b>	40 (17%)	197 (83%)	237	36 (24%)	116 (76%)	152
<b>2019</b>	41 (17%)	200 (83%)	241	32 (20%)	132 (80%)	164
<b>2018</b>	34 (16%)	177(84%)	211	32 (22%)	112 (78%)	144
<b>2017</b>	29 (13%)	199 (87%)	228	27 (20%)	106 (80%)	133
<b>2016</b>	36 (17%)	170 (83%)	206	32 (21%)	117 (79%)	149

La información muestra que, en el caso de la Universidad Nacional del Comahue, las nuevas inscripciones no presentan una variación significativa en el período reportado (10%). Mientras que, en las carreras de grado el porcentaje de participación femenina se mantiene estabilizado alrededor del 16%, en las tecnicaturas se observa una ligera variación del 21% en 2016 hacia el 24% en 2020. En ambas situaciones la población femenina se describe como subrepresentada.

Las Tabla 3 y Tabla 4 presentan datos sobre la cantidad de egresos mostrando que si bien la representación femenina es baja en relación a la masculina guarda relación con las tasas de ingreso.

Tabla 3. Cantidad de estudiantes que egresaron en carreras de computación de la UNC (2016-2020)

CANTIDAD DE EGRESOS UNC (2016-2020)						
Año	Ingeniería en Computación			Licenciatura en Ciencias de la Computación		
	Mujeres	Hombres	Total	Mujeres	Hombres	Total
2020	1 (6%)	16 (94%)	17	1 (11%)	8 (89%)	9
2019	0 (0%)	20 (100%)	20	2 (11%)	16 (89%)	18
2018	2 (12,5%)	14 (87,5%)	16	2 (12%)	15 (88%)	17
2017	1 (6%)	15 (94%)	16	8 (17%)	40 (83%)	48
2016	0 (0%)	18 (100%)	18	1 (4%)	26 (96%)	27

Tabla 4. Cantidad de estudiantes que egresaron en carreras de computación de la UNComa (2016-2020)

CANTIDAD DE EGRESOS UNComa (2016-2020)						
Año	Carreras de Grado			Técnicaturas		
	Mujeres	Hombres	Total	Mujeres	Hombres	Total
2020	0 (0%)	8 (100%)	8	2 (13%)	13(87%)	15
2019	2 (50%)	2 (50%)	4	4 (22%)	14 (78%)	18
2018	4 (57%)	3 (43%)	7	7 (39%)	11(61%)	18
2017	4 (50%)	4 (50%)	8	4 (16%)	21 (84%)	25
2016	4 (40%)	6 (60%)	10	6 (25%)	19 (76%)	25

Como muestran las Tabla 3 y Tabla 4, el panorama de estudiantes egresados no sólo preocupa en relación a la perspectiva de género sino también en cuanto al decaimiento del número de estudiantes que logra finalizar sus estudios superiores. Si bien, no hemos considerado el título intermedio de analista que otorgan las licenciaturas, éstas carreras muestran números desalentadores. Esto puede relacionarse a lo conocido como la fuga de jóvenes estudiantes a la industria del software con sueldos tentadores, en donde, no se les requiere la titulación ya que la demanda de recursos humanos, como presentamos al inicio del texto, y la gran oferta laboral a cubrir, abona a que estos números sean preocupantes para las casas de altos estudios.

Aunque la tasa de egreso femenina es menor a la masculina, se observa una diferencia poco significativa en términos de deserción discriminada por género. Por ejemplo la matrícula femenina ingresante a las tecnicaturas dictadas por la UNComa en el período 2014 - 2018, que se corresponde con los egresos 2016 - 2020, representan el 23.41% de las inscripciones y 22.71% de los egresos en el período 2016 - 2020.

Con el fin de adentrarnos en aquello que sucede en nuestras carreras universitarias, hemos realizado una encuesta a estudiantes que cursan materias de primer año, como Introducción a los Algoritmos en UNC e Introducción a la Computación en UNComa, para conocer un poco más sobre cómo aparece la perspectiva de género en los ingresos a estas carreras de computación. A continuación, retomamos las respuestas estudiantiles a la

pregunta **¿Por qué decidiste estudiar esta carrera? y cómo se vincula esto con las escasas referentes femeninas que existen en el área.**

### 3- Referentes en el área de las Ciencias de la Computación

#### 3a- Caso FaMAF (UNC)

Se recuperaron los datos de una comisión de la materia de primer año de la Licenciatura en Ciencias de la Computación (CC) que dicta la Facultad de Matemática, Física, Astronomía y Computación (FaMAF), en la cual se preguntó a su estudiantado, al inicio del año lectivo **¿Por qué decidiste estudiar esta carrera? (si no planeas hacer toda la carrera, por qué estás haciendo esta materia).** Del total de 111 inscriptos en 2022, 30 estudiantes son *mujeres* (27%), 80 *varones* (72%) y 1 persona respondió la opción *otro género* (1%). Analizando los datos de las mujeres y del género otro se encontraron respuestas en dónde se admitía más de una categoría de elección.

Estos datos arrojaron que el mayor porcentaje relacionado a la decisión de por qué estudiar Ciencias de la Computación está vinculado a un **interés por la computadora** (35%), luego aparece un **gusto para el desarrollo del software y la programación** con un 26%, la **salida laboral** con un 18% y en menor medida aparecen las respuestas están vinculadas al **gusto por la matemática** (6%), **querer aprender** (6%), por su **relevancia mundial** (6%) y por su **importancia** (3%).

Este mayor porcentaje ligado al interés por esta disciplina, presentaba además, la mención del estudiantado al acercamiento a personas que ya se vinculan con las ciencias de la computación.

*“Mi pareja es analista programador y siempre a final del día hablamos de cómo nos había ido a los dos, él siempre me hablaba de todo lo que había hecho y programado en diferentes lenguajes y un día me dio curiosidad y me interese por investigar de qué se trataban los lenguajes de programación y para que se usaban (...) (Estudiante femenina, FaMAF, UNC, 2022)*

*“Mis dos mejores amigos son programadores por lo cual siempre tuve conversaciones al respecto y al final, después de cursar Semiótica en comunicación relacioné las dos cosas y me atrapó la idea de hacer e interpretar lenguajes nuevos.” (Estudiante femenina, FaMAF, UNC, 2022)*

Incluso una estudiante se explaya en la encuesta mencionando: *“Siempre me gusto la computación pero cuando me egrese del secundario (2005) no era tan conocida la opción y termine decantando por hacer bioquímica en la facu de al lado. Me recibí y actualmente ejerzo. Con la pandemia dije "ya fue! hago todo lo que me gusta" y acá me ven.”*

De esta manera, el contacto con gente cercana a la disciplina, habilita representación sobre el área y permite conocer los alcances de las carreras de Computación. La poca información, como menciona la última estudiante, puede tener repercusiones como elegir

otras carreras por desconocer sobre la posibilidad de formalizar esos intereses en la computación.

Estos datos nos alertaron de la carencia en la mención de referentes mujeres vinculadas al oficio de la computación. Por ello, decidimos realizar un nuevo cuestionario con preguntas abiertas indagando sobre si “**Antes de anotarte en la carrera, ¿conocías a alguien que esté vinculado a las Ciencias de la Computación?**”. Por vía mail y a través del contacto de docentes, hemos distribuido la herramienta de recolección de datos y hemos conseguido las respuestas de 51 estudiantes de los cuales, 32 son varones, 16 mujeres, 2 respondieron género otro y una prefirió no responder.

Tabla 5: Respuestas a la pregunta: Antes de anotarte en la carrera, ¿conocías a alguien que esté vinculado a las Ciencias de la Computación?.

Antes de anotarte en la carrera, ¿conocías a alguien que esté vinculado a las Ciencias de la Computación?.			
Género	Sí	No	Total
Mujeres	6	10	16
Otro	2	0	2
N/R	0	1	1
Varones	17	15	32

Del total de respuestas afirmativas de las mujeres (6), una no respondió y en las 5 restantes mencionan conocer a personas de género masculino, aunque dos de ellas agregan que también conocen mujeres: “*Conocía aproximadamente 10 personas que cursaron/cursan la carrera, mi pareja (hombre cis 24 años) y sus amigos (mayoría hombres cis de entre 23 y 28 años y dos mujeres cis de entre 22 y 27 años) y un amigo de toda la vida (hombre cis 21 años)*” y “*Mi prima se egreso hace un par de años y ahora es profesora (...)*”. Incluso del número de varones que respondieron conocer personas vinculadas al área (17), sólo 4 mencionan además conocer a mujeres, las cuales están ligadas a la docencia (profesoras de secundaria) o aún son estudiantes. No se registran respuestas de mujeres vinculadas a la industria del software. Este último dato, tiene sentido en relación a la escasa participación de mujeres en el área laboral tecnológica, tal como se presentó al inicio del escrito.

### 3.b Caso Facultad de Informática (UNComa)

Consultamos a estudiantes que cursan la materia Introducción a la Computación, se trata de una de las materias del primer cuatrimestre del primer año de estudio de las carreras Licenciatura en Ciencias de la Computación (LCC), Licenciatura en Sistemas de Información (LSI) y Profesorado en Informática (PI) dictadas por la Facultad de Informática. La consulta incorporó una dimensión relacionada a la identificación de personas que se ubicaron como referentes para con los procesos de elección de profesiones vinculadas a la computación.

El número de estudiantes que respondieron a la consulta es 50, entre quienes 14 (28%) se auto perciben con el género femenino, 34 (68%) con el género masculino, 1 (2%) con otro género y 1 (2%) prefirió no decirlo. Como instrumento de indagación se adoptó un cuestionario en línea distribuido por medio de los canales de comunicación habituales para la cátedra.



A la pregunta “**Antes de anotarte en la carrera, ¿conocías a alguien que esté vinculado o vinculada a las Ciencias de la Computación?**” 16 (32%) estudiantes manifestaron conocer a alguna persona relacionada a la computación que resultó relevante en la definición de su proyecto de vida profesional. Entre las respuestas afirmativas, 9 (56%) estudiantes expresan haber conocido a personas del género masculino, 4 (25%) informan que conocieron tanto a varones como a mujeres y 3 no aportan este dato. Mientras el 81% menciona a personas de género masculino y el 25% hace referencia a mujeres, nadie nombra sólo a mujeres.

Tabla 6: Respuestas a la pregunta: Antes de anotarte en la carrera, ¿conocías a alguien que esté vinculado a las Ciencias de la Computación?

<b>Antes de anotarte en la carrera, ¿conocías a alguien que esté vinculado o vinculada a las Ciencias de la Computación?</b>			
<b>Género</b>	<b>Sí</b>	<b>No</b>	<b>Total</b>
Mujeres	4	10	14
Otro	1	0	1
Prefiere no responder	0	1	1
Varones	11	23	34

Al revisar las respuestas, es posible identificar tres grupos de personas que resultaron relevantes al momento de elegir una carrera computacional, en este sentido el 46% refiere a familiares, el 38% docentes de sus escuelas secundarias y el 15% a amistades.

*“Mis tíos (tía y tío maternos) son programadores. Cuando mostré interés en la computación fueron de los primeros en ponerse felices por mí, darme consejos, recursos para empezar a aprender sola.” (Estudiante autopercebida femenina, Facultad de Informática, UNComa, ingresante 2022)*

*“Un profesor del secundario que nos enseñó algunas cosas básicas sobre elementos de la computadora y programación con PSeInt.” (Estudiante masculino, Facultad de Informática, UNComa, ingresante 2022). “Un profesor de secundaria, y me hablo un poco del tema para elegir que carrera elegir” (Estudiante autopercebido masculino, Facultad de Informática, UNComa, ingresante 2022)*

*“La persona que me hizo conocer la carrera es un ex alumno de la secundaria, íbamos a la misma secundaria y él fue quien me incursionó en el mundo de la programación y la computación” (Estudiante autopercebida femenina, Facultad de Informática, UNComa, ingresante 2022)*

En este marco, se registran dos respuestas referidas a mujeres que trabajan en el ámbito las tecnologías, aunque ninguna respuesta menciona sólo mujeres, “*Mis tíos (tía y tío maternos) son programadores*” y “*Mi papá y algún que otro compañero/a de él, trabajan programando en un banco*”. El 68% de las respuestas no se refiere explícitamente a personas

vinculadas al campo laboral de la computación, una proporción importante de este grupo corresponde a docentes de escuelas secundarias.

Los datos analizados por ambas universidades arriban a resultados similares, evidenciando una escasa cercanía de mujeres referentes en el área de la computación y las tecnologías. Situación que repercute en las representaciones que se pueden construir sobre las mujeres y el área, y fomenta la permanencia de la brecha de género. En el siguiente apartado nos adentramos en observar qué visibilidad presenta nuestro estudiantado con respecto a las desigualdades de género en sus carreras.

#### **4- Desigualdades de género en las carreras**

##### **4.a Caso FaMAF (UNC)**

En el cuestionario aplicado también se les preguntó si identificaban desigualdades de género en la materia o en la carrera. De las 16 mujeres, 13 respondieron que no registrar diferencias de género, ni en la materia ni en la facultad, y sólo tres mencionaron observar desigualdades en la cantidad de mujeres en la carrera y en la materia. Incluso una de ellas menciona que no identifica diferencias pero sí registra que hay una desigualdad en la matrícula, lo que permite avizorar un estado de naturalización en la existencia de carreras más masculinizadas que otras, cuando en realidad no deja de ser una construcción social y por lo tanto atravesado por sesgos (Foures y Aguirre, 2020). Lo mismo para con las 32 respuestas de los estudiantes varones, de las cuales 30 no reconocen disparidad y solo 2 mencionan una mayor cantidad de matrícula masculina con respecto a otros géneros.

Aparecen algunos comentarios que comienzan a vincular el género con el conocimiento, por ejemplo, *“Hay muchos hombres y dan por sabido que las mujeres también saben”*, *“Algunos comentarios machistas sobre que por ser mujer me cuesta más, desconozco más e incluso que no pertenezco a este tipo de carreras “para hombres”. Son comentarios de alumnos.”* Incluso, en aquellos que se reconocen con el género otro o prefieren no responder, sólo una persona mencionó identificar desigualdad:

*“Un poco noto que las mujeres tienen más vergüenza al preguntar y las veces que preguntan noto que los docentes simplifican sus respuestas. (En la carrera) Además, he notado que en instancias orales le ponen nota más alta a hombres aunque la mujer haya tenido el mismo tipo de error, o ningún error en absoluto. Supongo que es por algún tipo de sesgo machista.” (Estudiante de género otro, FaMAF, UNC, 2022)*

Morgade (2005) en sus investigaciones sobre representaciones acerca del propio rendimiento y del rendimiento del otro sexo en las áreas de Ciencias Exactas y Matemáticas encuentra una bipolaridad de género en la relación con el conocimiento escolar fundamentada por la naturalización de creer que los varones, por naturaleza,

“son más inteligentes”, “les resulta más fácil”. Contrariamente a las mujeres, a las cuales, la naturaleza no las acompañaría y para tener éxito en la escuela tienen que quebrar ese “orden natural”. De esta manera, se atribuiría a las mujeres dificultades ligadas a factores personales como “costarle más” a diferencia de los varones, donde no se pone en duda ni sus capacidades ni sus habilidades (Arcanio, 2008).

#### **4.b Caso Facultad de Informática (UNComa)**

Como en el Caso FaMAF, el cuestionario aplicado consulta acerca de si identifican desigualdades de género en la materia o en la carrera. Entre las 50 respuestas recogidas, sólo una persona, que se percibe de género masculino, manifiesta que en la carrera y en la materia existen marcadas desigualdades de género.

*“Sí, la gran mayoría de alumnxs son hombres, y el comportamiento en espacios de interacción virtuales dejan que desear. Una de las primeras cosas que noté al empezar la carrera es que hay bastante misoginia y homofobia ‘disfrazada’ de comedia. Hay comentarios hechos como bromas que en mi opinión podrían incomodar a ciertas personas. No lo he percibido tanto en ámbitos presenciales” (Estudiante autopercebido masculino, Facultad de Informática, UNComa, ingresante 2022)*

Otra estudiante, refiere a un imaginario social acerca de la preponderancia masculina en las carreras computacionales y que esa situación le producía temor. A la pregunta En el cursado de la carrera, ¿identificás alguna desigualdad de género?, responde:

*“No, de hecho es algo a lo que le tenía miedo. Por más que tenía de ejemplo a mi tía programadora, sabía que es un campo dominado por hombres. Ella me lo advirtió, y se nota al ver los inscriptos. Pero llegar y ver que hay tanto profesores como profesoras dando teoría y ayudantes me animó bastante.” (Estudiante autopercebida femenina, Facultad de Informática, UNComa, ingresante 2022)*

En relación a la disparidad de género en las carreras, aunque la preponderancia masculina resulta evidente en la composición de la matrícula, el 98% de la población estudiantil consultada no lo identifica como un problema de inequidad y el 8% manifiesta explícitamente que considera que se trata de una situación normal.

*“Principalmente creo que es más una cuestión de preferencias” (Estudiante autopercebida femenina, Facultad de Informática, UNComa, ingresante 2022)*

*“Hay una buena cantidad de mujeres en las clases” (Estudiante autopercebido masculino, Facultad de Informática, UNComa, ingresante 2022)*

*“Si bien existe una ‘brecha de género’ no creo que sea relevante en lo absoluto, simplemente las preferencias de ciertos grupos de personas son distintas, no hay nada más allá que eso.” (Estudiante autopercebido masculino, Facultad de Informática, UNComa, ingresante 2022)*

Finalmente, se observa un grupo de respuestas que demuestran incomodidad o enojo ante el abordaje de la cuestión de género en el ámbito del dictado de la cátedra. Se recogieron 5 (10%) respuestas en este sentido.

*“Me hace pensar que quieren meter ideología de género de cualquier manera, en una carrera que no tiene absolutamente nada que ver con esos temas.” (Estudiante autopercebido masculino, Facultad de Informática, UNComa, ingresante 2022)*

*“En esta carrera, no hay ninguna desigualdad de género, como dije en la pregunta anterior, siendo esta carrera una de las que menos materias sociales tiene, y donde se debería trabajar ese tema.” (Estudiante autopercebido masculino, Facultad de Informática, UNComa, ingresante 2022)*

La información recogida muestra que la baja presencia femenina en carreras relacionadas a la computación y en la industria es un problema sin resolver. Múltiples factores influyen la elección y persistencia de mujeres en carreras computacionales, entre estos aspectos las condiciones de equidad durante los procesos formativos y las experiencias previas en ciencias de la computación se ubican como predictores importantes en persistencia de las mujeres en las carreras relacionadas con la computación (Weidler-Lewis, 2019).

## **5- Algunas reflexiones para pensar en conjunto**

En este escrito pretendemos mirar los datos vinculados a los ingresos y egresos estudiantiles en carreras de informática con perspectiva de género, para compartir con las comunidades educativas y poder pensar, a partir de allí, movimientos que ayuden a lograr una equidad. Aclaramos que no aspiramos necesariamente a una igualdad en cantidad de porcentajes de participación, sino que estas diferencias no sean por concepciones estereotipadas sobre roles asignados culturalmente a lo masculino y femenino, vinculados al quehacer tecnológico y digital.

Estas diferencias no sólo se generan en las elecciones universitarias sino que suelen reproducirse en los vínculos educativos. En algunas respuestas analizadas, se observa un sesgo de género relacionado al conocimiento cuando se menciona la existencia de distintos saberes que no apelan a la singularidad ni trayectoria estudiantil, sino a cuestiones de género.

Incluso esto también se observa en algunas respuestas docentes en interacciones en las clases: menor calificación o simplificación de respuestas para las mujeres. Al respecto, Paloma Moreda (2021) menciona que, la perspectiva de género permite a quien ejerza la docencia, prestar atención a las dinámicas de género que tienen lugar en el entorno de aprendizaje y adoptar medidas que aseguren que se atiende a la diversidad de los y las estudiantes. “Al fin y al cabo, incorporar el principio de igualdad de género no es solo una cuestión de justicia social, sino de calidad de la docencia” (p. 6).

Brindar visibilidad a las mujeres que actualmente se están desarrollando en la industria de software puede ser un movimiento que alienta a que más mujeres se puedan reconocer como partícipes de esa comunidad laboral. Es por ello, que presentamos estos datos para que, en conjunto, podamos pensar propuestas micro y macro para contrarrestar estas desigualdades de género, ya sea desde el interior de nuestras prácticas docentes hasta políticas de divulgación y acción.

## 6- Bibliografía

- Arcanio, M Z (2008). Las construcciones sociales de género y las trayectorias académicas. Líneas de análisis para pensar su articulación en el pasaje del nivel medio a la universidad. XV Jornadas de Investigación y Cuarto Encuentro de Investigadores en Psicología del Mercosur. Facultad de Psicología - Universidad de Buenos Aires, Buenos Aires.
- Berro, I. (2021). Mujeres en computación: una carrera con obstáculos. Tesis de Licenciatura en Desarrollo. Facultad de Ciencias Sociales. Universidad de la República (Uruguay).
- Cheryan, S. (2012). Understanding the paradox in math-related fields: Why do some gender gaps remain while others do not?. *Sex roles*, 66(3), 184-190.
- Comisión de la Condición Jurídica y Social de la Mujer-CSW61- (2017). "Empoderamiento económico de la mujer en el cambiante mundo del trabajo" ONU MUJERES. Sede Naciones Unidas. Nueva York. Recuperado de: [https://www.unwomen.org/sites/default/files/Headquarters/Attachments/Sections/CSW/UNW\\_CS61\\_Brochure\\_SP.pdf](https://www.unwomen.org/sites/default/files/Headquarters/Attachments/Sections/CSW/UNW_CS61_Brochure_SP.pdf)
- Echeveste et al, 2021 Echeveste, M. E., Gómez, M. J., Martínez, C., y Benotti, L. (2021) La escuela y la brecha de género en la enseñanza de las Ciencias de la Computación. Jornadas Argentinas de Didáctica de las Ciencias de la Computación. Recuperado de: [https://www.researchgate.net/profile/Marcos-Gomez-5/publication/359931576\\_La\\_escuela\\_y\\_la\\_brecha\\_de\\_genero\\_en\\_la\\_ensenanza](https://www.researchgate.net/profile/Marcos-Gomez-5/publication/359931576_La_escuela_y_la_brecha_de_genero_en_la_ensenanza)
- Foro Económico Mundial -WEF- (2018) The future of jobs report. Suiza: Foro Económico Mundial.
- Foures, C. y Aguirre, J. (2020). Aportes para el análisis de la feminización de carreras históricamente masculinizadas. Destejiendo prejuicios de género. *Revista de las Ciencias sociales*; 7 (1); 349-358
- Fraillon, J., Schulz, W., Friedman, T., Ainley, J., & Gebhardt, E. (2015). ICILS 2013 Technical Report. International Association for the Evaluation of Educational Achievement. Herengracht 487, Amsterdam, 1017 BT, The Netherlands.

- Henn, 2014. When Women Stopped Coding. Planet Money. <https://www.npr.org/sections/money/2014/10/21/357629765/when-women-stopped-coding>
- Morgade, G. (2005): “Lectura de género y procesos educativos”. En: Revista Criterio No 2309. Año 78. [www.revistacriterio.com.ar](http://www.revistacriterio.com.ar)
- Moreda, P. (2021). Guías para una docencia universitaria con perspectiva de género. Ciencias de la Computación.
- ONU (2020) Las mujeres en ciencias, tecnología, ingeniería y matemáticas en América Latina y el Caribe. Montevideo. Recuperado de <https://www2.unwomen.org/-/media/field%20office%20americas/documentos/publicaciones/2020/09/mujeres%20en%20stem%20onu%20mujeres%20unesco%20sp32922.pdf?la=es&vs=4703>
- Pineda, E., y Gonzalez, C. (2016). Networking Skills in Latin America , White Paper, Cisco IDC Skills Gap.
- Sadosky, F. (2013). Y las mujeres...¿ Dónde están? Primer estudio de la Fundación Dr. Manuel Sadosky sobre la baja presencia femenina en informática.
- Soria-Barreto, K., & Zuniga-Jara, S. (2020). Tendencias en la paridad de género en carreras universitarias de computación en Chile: 2004-2018. *Formación universitaria*, 13(5), 35-44.
- UNESCO (2018). “Resumen sobre género informe de seguimiento de la educación en el mundo”. Recuperado de <https://unesdoc.unesco.org/ark:/48223/pf0000261945>
- Wigdor, G. B. (2016). Aferrarse o soltar privilegios de género: sobre masculinidades hegemónicas y disidentes. *Península*, 11(2), 101-122.)
- Weidler-Lewis, J., DuBow, W., Kaminsky, A., & Weston, T. (2019). Supporting women’s persistence in computing and technology: A case for compulsory critical coding?. *Information and Learning Sciences*.

## **La incorporación de las Ciencias de la Computación en el currículum: ¿la transversalidad como alternativa?**

Magdalena Garzón

Fundación Sadosky

[mgarzon@fundacionsadosky.org.ar](mailto:mgarzon@fundacionsadosky.org.ar)

### **Resumen**

Este estudio analiza el debate actual para organizar los contenidos de las Ciencias de la Computación (CC) en la escolaridad obligatoria. Mientras algunos sostienen que es necesario establecer un espacio curricular propio, otros abogan por un enfoque transversal que integre estos contenidos en todos los espacios curriculares vigentes. Para ello, se consideran diversas dimensiones. La literatura educativa especializada no niega la necesidad de las disciplinas para un abordaje transversal sino por el contrario requiere que todas o varias disciplinas aporten su mirada y dimensión de análisis. Los enfoques que consideran a la Alfabetización digital (referida no solo a la formación de usuarios sino también a la comprensión de los saberes pertinentes para producir e intervenir tecnología de manera ética) como una competencia del siglo XXI, tampoco deberían traducirse en la ausencia de las disciplinas, ya que el desarrollo de este tipo de competencia necesariamente se apoya en los conocimientos disciplinares. Por otro lado, la transversalidad de un tema o de una competencia clave ha enfrentado de forma sistemática dificultades para traducirse en el diseño de prácticas de enseñanza. En este estudio se entrecruzan los conceptos y miradas del campo educativo sobre la transversalidad con los saberes de computación requeridos para una ciudadanía plena, con el propósito de echar luz frente a la toma de decisiones de los diseños curriculares.

Palabras clave: currículum, computación, alfabetización, transversalidad, competencias, computación.

### **Introducción**

Existe cierto consenso sobre la necesidad de incorporar a la educación obligatoria saberes propios de las Ciencias de la computación. En la actualidad estos saberes exceden el enfoque que predominó en la década del 2000 sobre el uso de TIC. La pandemia del COVID-19 subrayó el momento histórico de intensificación del acceso, desarrollo y apropiación de las tecnologías digitales y dejó al descubierto las brechas digitales existentes, no solo socioeconómicas, sino también educativas, geográficas y de género.

Muchos países del mundo se encuentran frente a las mismas definiciones. En un reciente estudio Martínez y Borchardt (2021) relevaron las perspectivas pedagógicas de la introducción de las Ciencias de la Computación en 10 países. Respecto de la organización curricular, concluyen que la mitad de los países relevados eligió un enfoque transversal o integrado, mientras que la otra mitad

definió una materia con contenidos específicos o incluye estos contenidos en una materia ya existente pero reformulada para abordar las CC.

En la actualidad una gran cantidad de jurisdicciones argentinas se encuentran con el desafío de resolver de qué modo garantizarán que las y los estudiantes puedan acceder al mundo digital y participar de forma reflexiva y crítica. Los Núcleos de Aprendizajes Prioritarios de Educación Digital, Programación y Robótica, (NAP EDPR) (Resolución CFE N° 343/2018) dieron un impulso al establecer la obligatoriedad de incorporar determinados saberes en todos los niveles de la escolaridad. Sin embargo, la falta de claridad conceptual sobre los temas que involucran el uso de TIC, la informática, la educación digital, la programación y la robótica, reportada por la Unesco (2019), no contribuye a la toma de decisiones para su implementación. La “ensalada epistemológica que la Informática trajo a la Educación”, como la refiere Schapachnik (2021) no se limita a discusiones académicas sino que influye en el discurso de la política educativa (Albarello, Hafner, 2019).

La multiplicidad de aspectos a resolver para su efectiva implementación no son sencillos, involucran: definir qué saberes son necesarios para lograr los propósitos delineados en las normativas, atender a la formación docente inicial y continua, establecer el modo en que se introducirán en el currículum; todo ello sin desconocer la estructura escolar.

Por el momento las jurisdicciones de nuestro país que avanzaron en sus reformas curriculares lo han hecho como disciplina específica en el nivel secundario en el marco de la reforma de este nivel. Neuquén (2018) y CABA (2015) creando un nuevo espacio curricular. Otras jurisdicciones están llevando adelante propuestas aún no universalizadas, como Tucumán y Chaco que reformularon los contenidos de Tecnología en una muestra de escuelas vinculadas a la iniciativa PLaNEA, UNICEF (2017). Y otras, van creando escuelas secundarias experimentales intensificadas en tecnología, como las escuelas PROA de Córdoba (2014) vinculadas al desarrollo de software o el modelo de la escuela de Innovación en Misiones (2019) orientada a la programación y la robótica educativa que en 2021 inició su escalabilidad, entre otras.

### **La integración en el currículo de modo transversal**

Los NAP EDPR no prescriben un modo de incorporar los saberes en la escuela, no obstante, en sus fundamentos y en las orientaciones pedagógicas dejan la puerta abierta a la integración en un “proceso de transversalidad” (Cuadernillo NAP EDPR, página 8).

El concepto o la idea de lo transversal en el escenario educativo, no es algo nuevo. En la Argentina la reforma curricular de los años noventa introdujo esta tipificación y en ese entonces suscitó debates, polémica y críticas (Ferreyra, 2004; Daino y Rojas, 2006), como la de vaciar a la escuela de contenido, la de descentrar el rol de la escuela en tanto institución que transfiere contenidos relevantes a las nuevas generaciones (Dussel, 2020) o la de devaluar la enseñanza disciplinar en su estructura conceptual y en la profundidad de sus desarrollos teóricos (Pardo, 2021). Kysilka, M L. ya en el año 1998 analizaba esta posibilidad en el currículum de Estados Unidos y concluía: *El lenguaje de la integración curricular es confuso y genera incertidumbre y preocupación sobre el potencial del currículo integrado para tener un impacto positivo en las escuelas. [...] Por el momento parece que*



*la integración significa lo que sea que alguien decida que significa, siempre y cuando haya una "conexión" entre áreas de contenido y habilidades que previamente fueron separadas (1998: 197-198).*

Para comprender la perspectiva de la transversalidad es necesario enmarcarla en un modo de organización curricular de los contenidos. Entre muchos otros que fueron surgiendo desde Dewey (1916), Kilpatrick (1918), Oberholtzer (1937), Squires (1972), Vars (1969, 1987) y Beane (1993), cada uno ha propuesto su terminología para nombrar estos modelos. En este camino aparecieron las propuestas más diversas de integración curricular que van desde las materias separadas (como las conocemos en su forma más tradicional) en la hipótesis de que no hay ninguna integración, modelos basados en dos o más disciplinas que abordan en simultáneo un tema pero cada una en forma independiente, modelos interdisciplinarios en los que dos o más disciplinas abordan en conjunto un tema o problema multidimensional, hasta modelos de integración total, como la transversalidad. Todos ellos, surgen en la búsqueda de modos en que los y las estudiantes puedan establecer una conexión entre lo que aprenden en la escuela y la información, las habilidades y los conocimientos que utilizan en situaciones de la vida real (Kysilka, M., 1998).

Como destacan Martínez y otros (2021) en muchos países, inclusive el nuestro, la transversalidad de saberes en la escuela se ha abordado en dos planos (Daino y Rojas, 2006):

- A. como temas que abordan demandas y problemáticas sociales que requieren del aporte de varias disciplinas, de la institución o la comunidad escolar;
- B. como competencias o habilidades generales que atraviesan el acto de enseñar.

### **Los contenidos transversales y su implementación en la escuela**

En su sentido etimológico estricto “transversal” es aquel contenido que “atraviesa” o “impregna” todo proceso de enseñanza-aprendizaje (Daino y Rojas, 2006). Esta simple definición para el lenguaje común genera mucha incertidumbre cuando es utilizada como instrumento de diseño curricular y aporta muy poca precisión en el momento de su implementación. Por esta razón, pocas veces se traduce en prácticas escolares concretas (MEN Santa Fe, 1998).

Gutman y Siede (1995) advierten que la noción de transversalidad tiende a cuestionar los encasillamientos, por ese motivo es tan difícil enmarcarla en tipos o categorías. Estos autores plantean que las perspectivas transversales enfocan su mirada en algunos aspectos básicos de la formación integral de las personas, en la transmisión crítica de valores y la adopción de normas y pautas de conducta. En este sentido, son perspectivas globales que conectan a la escuela con las necesidades y preocupaciones de la sociedad y de la comunidad particular en que ella se inserta.

Con esta perspectiva, proponen pensar en múltiples modos en que los contenidos se “cruzan en el camino de la escuela”, por ejemplo:

- Temas emergentes producto de una situación o hecho puntual, que necesitan ser abordados desde una perspectiva que permita significarlos de algún modo. La guerra de Rusia y Ucrania, el COVID-19, los incendios forestales, el último campeonato mundial de fútbol.
- Problemáticas contemporáneas de nuestro país y del mundo en tanto la escuela es caja de resonancia de situaciones que estallan en un momento dado y entran a la escuela de la mano de niños/as, madres/padres y maestros/as: temáticas como la desocupación, la crisis de los modelos familiares.
- Temáticas que requieren una concientización permanente: la educación vial, la educación para el consumo, la educación sexual integral, la educación ambiental, la multiculturalidad.
- Contenidos que operan sobre las concepciones didácticas y en los que la estrategia de enseñanza opera sobre los contenidos: la enseñanza de los derechos humanos, por ejemplo, compromete las pautas de la relación entre docentes y alumnos.

De estas concepciones se desprende que **no todos los temas son plausibles de ser transversalizados, solo aquellos que cumplen determinadas condiciones.**

Por su parte Kysilka (1998) aporta claridad sobre el modo en que un modelo de integración curricular estrictamente centrado en temas transversales, se debería implementar en la práctica. Por ejemplo, la institución puede seleccionar la ética como tema. Entonces, cada docente, dentro de su propia disciplina, abordará la ética en función de la materia. Esto podría significar discutir el plagio en la clase de literatura mientras los estudiantes preparan un trabajo de investigación, analizar las decisiones tomadas por los políticos en una clase de historia, establecer reglas de comportamiento deportivo adecuado en las clases de educación física, concentrarse en la censura de la obra artística en las clases de bellas artes, debatir cuestiones de ingeniería genética en biología o la inteligencia artificial en tecnología. Las disciplinas permanecen intactas, el contenido de las disciplinas no se modifica, pero cada profesor pone de relieve el tema mientras trabaja con sus alumnos en los contenidos que deben aprender. En este sentido, los temas constituyen la base del plan de estudios y las disciplinas utilizan los temas para enseñar conceptos, temas e ideas específicas dentro de sus espacios curriculares.

En síntesis, en términos de la organización de los contenidos en el currículum, **el tratamiento de un contenido transversal requiere que todas o varias disciplinas aporten su mirada, su dimensión de análisis a un asunto que atraviesa a toda la institución escolar.**

### **El problema de la transversalidad en una institución compartimentada**

Uno de los problemas para que los contenidos transversales se traduzcan en prácticas escolares sistemáticas, aparece cuando está presente la idea de que *refieren a contenidos que no merecen un espacio curricular propio porque lo importante a enseñar son los conocimientos provistos por las*

*disciplinas. Así, los transversales pertenecen a todas las áreas curriculares y a ninguna en particular.* (Daino y Rojas, 2006:92). La responsabilidad de abordar estos contenidos transversales, en el contexto escolar no queda clara de quién es. Aún cuando se distinguen de otros contenidos por poseer una peculiar especificidad e importancia, el hecho de ser contemplados desde un punto de vista transversal, los hace aparecer difuminados, imprecisos o como novedades que se suman al trabajo escolar para una ocasión especial o como contenidos a añadir sobrecargando los programas (Moreno, 2009).

Terigi (2008) explica por qué las iniciativas transversales no permean en las escuelas secundarias a partir de lo que denomina el “trípode de hierro”: la clasificación de los currículos, el principio de designación de los profesores por especialidad graduados de un sistema formador que se organizó según la misma lógica y la organización del trabajo docente por horas de clase. La autora explica que estas características conforman un **patrón organizacional** que es muy difícil de modificar y que pueden echar luz sobre las dificultades que sistemáticamente han enfrentado los intentos de reforma. En función de la temática a abordar tampoco es posible contar con la **formación necesaria** en todo el cuerpo docente (esto sucede especialmente en el nivel primario pero tampoco está exento el profesorado secundario). Terigi reflexiona al respecto: *se ha procurado la incorporación de nuevas temáticas al currículum, pero la falta de docentes formados en esas temáticas llevó a que la novedad de los contenidos se viera diluida* (2008:29).

Por otro lado, pocas veces se advierte sobre cuál es la **diferencia** que introduce el término transversalidad frente a otras formas de integración de disciplinas. Hay quienes definen los contenidos transversales como los contenidos que no pueden ser tratados desde una única disciplina y que para su adecuado tratamiento se necesita de la concurrencia de los conocimientos provistos por diversas disciplinas. Entendida de esta manera el significado de la transversalidad se confunde con la **interdisciplinariedad**. (Daino y Rojas, 2006).

Es así como a menudo, el abordaje transversal como solución para incorporar nuevos contenidos, no resulta efectivo en la práctica.

### La transversalidad como competencia o habilidad

Según Rogiers (2000) las competencias apelan a la movilización de un conjunto de recursos (conocimientos disciplinares específicos, saberes de experiencia, automatismos, saber-hacer de diferentes tipos, etc.) y tienen un carácter “finalizado” para cumplir una función social. Se ejercitan o ponen en juego frente a una cantidad de situaciones determinadas que le otorgan contexto y una disposición. Las situaciones corresponden frecuentemente a problemas específicos ligados a una disciplina y son un aspecto central en el enfoque por competencias. La enseñanza que busca promover el desarrollo de una competencia brinda a las y los estudiantes la ocasión de ejercitarla. Una actividad de investigación, un problema complejo para resolver, un trabajo de producción personal son situaciones de integración de aprendizajes que en función del contexto disciplinar requerirán movilizar un conjunto de conocimientos de la disciplina y capacidades generales. Entre los ejempl

que propone el autor podemos encontrar la competencia de *llevar a cabo una investigación en ciencias* (2000:114), que requiere movilizar conocimientos específicos de la disciplina y seguir determinados procedimientos propios del conocimiento científico.

Adoptar un enfoque transversal de competencias o habilidades en la escuela, involucra varias dimensiones: la institucional, la comunidad, las áreas, y por último el aula. En un intento de aclarar la incertidumbre reinante, un interesante documento de trabajo elaborado por el Ministerio de Educación de Santa Fe (1998) argumenta que las habilidades transversales en sentido estricto responden a esta tipología cuando:

- se requieren, como condiciones básicas, para el aprendizaje de los contenidos de todas y cada una de las áreas curriculares;
- cada área curricular proporciona contextos que los conectan con experiencias concretas;
- su aprendizaje debe ser asumido como responsabilidad compartida por todos los miembros de la comunidad escolar y propiciado a través de estrategias acordadas que involucren las actividades en las que participan los alumnos, sean de carácter curricular o institucional;
- su aprendizaje compromete un vínculo estrecho entre la escuela y la comunidad e involucra una toma de posición frente a prácticas y costumbres de la comunidad.

En este contexto, la transversalidad de saberes en la escuela, abordado como competencias o habilidades generales que atraviesan el acto de enseñar, tradicionalmente se asoció a la comunicación oral y escrita, el pensamiento reflexivo y crítico y los modos de convivencia. Todas las materias escolares abordan estos conjuntos de habilidades ya que *donde hay enseñanza y aprendizaje hay lengua, pensamiento y convivencia* (Gobierno Santa Fe, 1998).

Los marcos conceptuales que surgieron a fines de la década del noventa, en torno a las habilidades para el siglo XXI (la *Alianza para el aprendizaje del siglo XX*, los Estándares de *ISTE*, el marco *EnGauge 21st century skills* de Meteri Group, las competencias del grupo *ACT21* liderado por la Universidad de Melbourne, las *Competencias clave* de la OCDE, entre otros) introdujeron otro tipo de habilidades vinculadas a la información, los medios y la tecnología donde se inscribe a la alfabetización informacional, la alfabetización en medios y la alfabetización digital.

El lenguaje de competencias volvió a emerger en la última década, de la mano de instituciones internacionales, como la OCDE y su propuesta de Brújula de aprendizaje 2030 (OECD, 2021), la Comisión Europea y su iniciativa Europa 2020, entre otros.

Estas iniciativas, que destacan las habilidades, derivadas posteriormente en competencias, se instalaron en los discursos de reformas curriculares “innovadoras” y de alguna manera contribuyeron a la pretensión de escindir ciertas competencias en detrimento de los contenidos y prácticas de enseñanza donde se desarrollan. Al respecto, es interesante el análisis que realiza Inés Dussel (2020) del currículum escolar en ocho países, entre ellos la Argentina. En ellos, identifica la presencia de habilidades cognitivas como objetivos de nuevas materias escolares o como competencias

transversales, pero alerta que estas distan de ser tan disruptivas como sugieren las exaltaciones mediáticas que las promueven y tampoco están tan consensuadas como prescriben las organizaciones internacionales, ya que denotan diferentes prioridades, luchas políticas divergentes y tradiciones pedagógicas singulares.

Rogiers alerta sobre el riesgo de centrar la enseñanza únicamente en competencias [...] *ello desembocaría en una enseñanza utilitaria, profesionalizante a ultranza. Desde el momento en que se adopta una línea de estudio orientada a las competencias, conviene estar atento a ello.* (2000:112).

### **El problema de separar los contenidos de las habilidades**

Desde el 'movimiento transversalista' se critica a las materias escolares *por su incapacidad para dar lugar a las competencias del siglo XXI y por estar desconectadas de los modos contemporáneos de producción de conocimiento que funcionan de forma transversal y creativa, [por formar parte de] una cultura que valora una pedagogía transmisiva, que no tienen en cuenta las necesidades y motivaciones de los alumnos y prescinden de los problemas de la vida real.* (Dussel, 2020).

Rogiers ya en el 2000 llamaba a superar esta "bipolarización reductiva" entre la adquisición de conocimientos (de los "disciplinarios") versus la adquisición de los "saber-hacer generales" como argumentar, tener perspectiva, trabajar en equipo, buscar información (de los "transversalistas"). El autor sostiene que no es posible desarrollar las competencias sin una reflexión disciplinar. Como dice Perrenaud (2006), el problema del desarrollo de las competencias no tiene nada que ver con una disolución de las disciplinas en una vaga *sopa transversal*. Según Rogiers, la competencia tiene un carácter esencialmente disciplinario porque se dirige a la resolución de problemas ligados a una disciplina. Para ello se apoya necesariamente en los conocimientos de esa disciplina y se inscribe en el modo en que una disciplina organiza sus saberes. Sucede que al mismo tiempo, para resolver esos problemas, la competencia se apoya sobre un conjunto de "saber-hacer generales", capacidades que son transversales. Y aquí, cuando se intenta separar a la competencia del contexto en el que se pone en juego, comienzan los malos entendidos. *Ninguna capacidad existe en estado puro y toda capacidad se manifiesta a través de la aplicación de los contenidos* (Meirieu, 1990:181). *La capacidad de clasificar, en sí misma no dice gran cosa. Se pueden clasificar lápices de tamaño y color, un conjunto de referencias bibliográficas [...] La mejor forma de desarrollar una capacidad es aprender a ejercitarla bajo contenidos bien diferentes los unos de los otros, y en particular, a través de distintas disciplinas* (Rogiers, 2000:107).

Kysilka (1998) ejemplifica cómo la destreza de pensamiento como la de clasificar en una clase de matemáticas de tercer grado, se relaciona con el conocimiento específico de las formas geométricas, unida a la elaboración de gráficos o a alguna otra destreza de "organización" para que los alumnos comprendan las características de las formas geométricas, mejoren sus destrezas de clasificación y desarrollen destrezas de organización mediante las cuales se pueda almacenar y utilizar la

información para su posterior consulta. Como sintetiza Savater en un reciente tuit: *Las capacidades son una forma de ordenamiento, no de saberes. Hay muchas formas distintas de ordenar una habitación, pero no se puede ordenar una habitación vacía* (@Savater, 2022).

La idea de separar las habilidades cognitivas, por importantes que resulten, de las tramas específicas de contenidos donde se desarrollan, no se tradujo en el diseño de prácticas de enseñanza sistemática ya que como dice la especialista en didáctica Mariana Maggio, *sabemos desde hace décadas que el aislamiento de las habilidades como tales y la dedicación programática a ellas no es garantía de su florecer* (Maggio, 2018:73). Por esta razón, conocer y comprender los conceptos que fundamentan y estructuran las prácticas, es un punto fundamental de la formación.

En iniciativas como la del K-12<sup>1</sup> (2016) o los estándares definidos por la CSTA (2017), podemos observar que las habilidades que definen en sus documentos dependen claramente de contenidos que forman parte de las Ciencias de la Computación.

En síntesis, **no se define el desarrollo de una habilidad o competencia por sí misma, ya que están contextualizados en las prácticas de la disciplina** que motivan su desarrollo en el proceso de enseñanza.

### **El caso del Pensamiento Computacional como competencia transversal**

Las habilidades que se destacan para definir lo que el pensamiento computacional involucra son:

- la descomposición de un problema en pasos sencillos;
- el reconocimiento de patrones;
- la abstracción;
- el diseño de algoritmos (entendidos como una serie de pasos que seguir).

Cuando estas habilidades aterrizan en el mundo educativo, se interpretan desde otro paradigma y generan gran empatía ya que hace siglos que en la escuela de diferentes modos y en variedad de contextos de forma sistemática se ejercitan estas habilidades. Reconocer patrones se hace en la clase de música al interpretar una partitura, se descomponen problemas en pasos sencillos en clases de aritmética, se siguen pasos en todo tipo de textos instructivos. Es decir, que la noción original se transforma al descontextualizar su referencia original. Es entendible que cuando estas habilidades de pensamiento están acompañadas por la palabra *computacional*, generen cierto desconcierto.

Por otro lado, el mensaje de que puede desarrollarse sin necesidad de abordar las ideas de la computación que lo fundamentan (Bonello y Schapachnik, 2020), no contribuyen a aclarar el panorama.

---

<sup>1</sup> El Marco de Ciencias de la Computación K-12, es una iniciativa liderada por la Association for Computing Machinery, Code.org, Computer Science Teachers Association, Cyber Innovation Center y National Math and Science Initiative en asociación con los estados y distritos de Estados Unidos.

La primera definición de pensamiento computacional que acuñó Jeannet Wing (2006) aludía a la puesta en práctica de habilidades propias de los informáticos para resolver problemas, pero como analiza Martínez (2020), la posterior definición de 2011 refiere a *la formulación de problemas y sus soluciones para que estas últimas estén representadas de forma que puedan llevarse a cabo de manera efectiva por un procesador de información* [sea una persona o computadora] *para desarrollar soluciones a problemas que pueden ser resueltos computacionalmente* (Wing, 2011).

El Pensamiento Computacional como habilidad o competencia no escapa al problema que genera separarlo de su trama de contenidos (lo que probablemente hubiera contribuido a echar luz sobre cómo favorecerlo en la educación obligatoria).

### **El problema de la transferencia**

En el enfoque de las competencias, subyace la idea de que es posible transferir una competencia desarrollada en un área de enseñanza a diferentes situaciones y contextos y por esta característica ameritan un abordaje transversal.

En países donde la propuesta es de saberes transversales, los contenidos específicos de Ciencias de la Computación se proponen como operaciones mentales generales y se asume que son transferibles entre las disciplinas. Por ejemplo el CIEB<sup>2</sup> de Brasil postula que aprender álgebra en la clase de Matemática contribuye al desarrollo del pensamiento computacional en tanto la resolución de problemas requiere el uso de lenguajes formales, fórmulas, tablas y gráficos y la identificación de patrones para establecer generalizaciones. Es decir se aborda la competencia general sin referencia a las situaciones problemáticas que permiten ponerla en juego, al decir de Rogiers (2000) ligadas a una disciplina y su estructura conceptual específica.

¿Encontrar un patrón en una secuencia de dibujos permitiría a las y los estudiantes en un contexto matemático identificar regularidades algebraicas en una secuencia numérica, o reconocer patrones en la secuencia de comandos de un programa computacional? Generalizar esta habilidad, requiere no solo dominarla en diferentes contextos, sino que representa una complejidad suficiente como para requerir un trabajo sistemático de explicitación de los aspectos comunes.

Martínez (2020) señala que algunas investigaciones han demostrado que las habilidades de pensamiento que se desarrollan en un dominio o disciplina, no siempre, ni de manera fluida, se transfieren para resolver problemas en otras disciplinas. En relación a la práctica de la programación, Clements (1999) destaca la necesidad de un andamiaje intencionado por parte de las y los docentes que ayude a sus estudiantes a establecer conexiones entre la programación informática y otras

---

<sup>2</sup> Centro de Innovación para la Educación Brasileira. <https://curriculo.cieb.net.br/>

experiencias académicas y cotidianas (Clements, 1999 citado en K-12 Computer Science Framework, 2016).

### La alfabetización digital desde un enfoque transversal

La alfabetización digital es un concepto que ha evolucionado con el tiempo. A comienzos del siglo XX el uso más extendido se refería principalmente a saber operar artefactos digitales (en un comienzo paquetes de ofimática, luego extendido a aplicaciones que permiten no solo consumir información, sino también crearla). A medida que la tecnología digital fue mediando en la mayoría de las dimensiones de nuestra vida diaria (nuestras relaciones interpersonales, el desarrollo profesional, el esparcimiento, el cuidado de la salud, por citar algunos ejemplos) el concepto se fue ampliando. La necesidad de comprender cómo funcionan los artefactos computacionales con los que interactuamos a diario, la posibilidad de producir e intervenir la tecnología de manera ética, fue cobrando fuerza como requisito para decodificar el mundo en el que vivimos.

El Estudio Internacional de Alfabetización digital y de la Información (ICILS por sus siglas en inglés) (Frailon y otros, 2019), busca dar respuesta a la pregunta ¿en qué medida están preparados los alumnos para estudiar, trabajar y vivir en un mundo digital? En el estudio realizado en 2013 se da cuenta de esta evolución. Según analiza Martínez (2020) el estudio ICILS medía la alfabetización digital, a partir de dos líneas: la primera referida al manejo y producción de información digital y la segunda referida a la transformación de la información digital vinculada a la alfabetización computacional que implicaba poder crear, transformar y compartir información y comprender el funcionamiento de una computadora.

En el estudio realizado en 2018 se agrega el constructo de Pensamiento computacional, que definen como *la habilidad de reconocer problemas del mundo real que son apropiados para la formulación computacional y para evaluar y desarrollar soluciones algorítmicas a esos problemas para que las soluciones puedan ser operativas con una computadora* (Frailon et al. 2019, p. 27). Los resultados del desempeño en el Pensamiento computacional, señalan que solo entre el 2 y 16% de los estudiantes alcanzan el nivel más alto que implica: explicar el valor de un sistema digital para resolver un problema, completar un simple árbol de decisión utilizando correctamente la lógica y la sintaxis, depurar un algoritmo para un problema complejo con una solución eficiente (por ejemplo, las tareas múltiples se resuelven mejor utilizando sentencias de repetición y condicionales). Entre sus conclusiones el estudio destaca que estas habilidades no se desarrollan solas a través de la exposición y el uso de dispositivos digitales, sino que requiere de una enseñanza sistemática.

Los NAP de Educación digital, Programación y Robótica (2018), aun cuando carecen de claridad respecto a qué contenidos son necesarios para desarrollar los saberes que promueve, en su génesis también abogan por una alfabetización digital, no solo en sentido instrumental sino desde una perspectiva de ciudadanía democrática que requiere de una apropiación crítica. Martínez y otros (2022) van un poco más allá, en su propuesta de un currículum emancipador de las Ciencias de la Computación que habilite a las y los estudiantes a vivir y desenvolverse de forma crítica, construir posiciones argumentadas y participar de las discusiones de nuestra sociedad actual.



Martinez y otros (2021) sostienen que *no podemos decir que gozamos de una escolaridad que nos permite decodificar el mundo en el que vivimos si un fenómeno que está tan presente en él y que cada vez media una cuota más importante de nuestras interacciones se nos presenta como una caja negra*. El impacto cultural y social de los desarrollos tecnológicos y las problemáticas contemporáneas vinculados a la ciudadanía y en particular a la ciudadanía digital, forman parte de nuestra vida cotidiana. El *ciberbullying*, el *grooming*, el *sexting*, el problema del poder, el control y la vigilancia en las redes, la gestión de la privacidad. Dilemas que involucran soluciones computacionales a nuestro alrededor, como la neutralidad de la red o los sistemas de Inteligencia Artificial que orientan la asignación de subsidios<sup>3</sup>, créditos bancarios o selección de personal y la consecuente preocupación por su ética (Martinez, V, 2022) e incluso sucesos como la caída masiva del servicio de WhatsApp<sup>4</sup>. La actual noción de alfabetización digital, que involucra la **comprensión de los saberes pertinentes para producir e intervenir tecnología de manera ética** obliga a traer al escenario la disciplina que permite comprender la tecnología computacional, que la aborda como objeto de estudio, que se ocupa de valorar, promover una mirada crítica, desarmar, desnaturalizar, historizar y transformar los artefactos computacionales.

Desde la perspectiva de la enseñanza para la comprensión (Boix Mansilla & Gardner, 2005) abordar una disciplina en la escuela requiere atender a cuatro dimensiones:

- La **dimensión del contenido**, refiere a que los estudiantes trasciendan las creencias intuitivas y construyan una red conceptual rica y coherente. En este sentido Pardo (2021) destaca que cada una de las distintas disciplinas posee conceptos estructurantes específicamente jerarquizados que requieren procesos de construcción singulares. Las CC, como disciplina escolar requiere abordar los principios y fundamentos de conceptos claves vinculados a la programación y algoritmia, a la infraestructura tecnológica (hardware, sistemas operativos y redes), a la recolección, procesamiento y análisis de datos (bases de datos, ciencia de datos y aprendizaje automático), a la seguridad.
- La **dimensión de los métodos**, refiere a que los estudiantes comprendan la forma de construir y validar conocimiento. Los modos de hacer computación, las prácticas en CC como reconocer y definir los problemas computacionales, desarrollar y utilizar abstracciones, crear artefactos computacionales (son algunas de las siete prácticas que identifica el marco conceptual propuesto por K-12 (2016)
- La **dimensión de los propósitos**, refiere a que los estudiantes consideren los intereses que guían la construcción de conocimiento de la disciplina. Esta dimensión en el ámbito de las CC

---

<sup>3</sup> Recientemente en Holanda, salió a la luz lo que la prensa llamó “The child benefits scandal” (El escándalo de las subvenciones por hijos a cargo) a raíz de un algoritmo de IA que utilizaba el gobierno holandés para determinar si las familias que recibían una subvención cometían fraude. El sistema identificó erróneamente a muchas familias a causa de perfiles raciales. En función de los resultados que devolvía el sistema se tomaban decisiones que iban desde desalojos hasta más de 1.100 niños sacados de sus hogares. <https://nltimes.nl/2021/10/19/1100-children-taken-homes-benefits-scandal-victims>

<sup>4</sup> En octubre de 2021 el servicio de Whatsapp tuvo una caída masiva que duró varias horas. Al respecto se recomienda la lectura del artículo ¿Qué podemos aprender de la caída de Whatsapp? de Fernando Schapachnik publicado en Sobre Tiza <https://www.sobretiza.com.ar/2021/10/04/que-podemos-aprender-de-la-caida-de-whatsapp/>

se vuelve indispensable dado el impacto que tiene la computación en el ejercicio de la ciudadanía. Favorecer una perspectiva crítica para visibilizar las innovaciones digitales, los intereses que motivan la creación de software y hardware.

- La **dimensión de las formas de comunicación**, refiere a que los estudiantes conozcan el sistema de símbolos propio y su forma de representar conocimiento. Las CC como disciplina científica trae al campo educativo la necesidad de expresar soluciones computacionales de alguna forma que pueda ser comprensible por otras personas y ejecutable automáticamente por una máquina.

## Conclusiones

Cuando nos referimos a las Ciencias de la Computación (CC), no estamos hablando de un tema o contenido puntual, tampoco de una única competencia que es menester desarrollar, sino de una disciplina que posee una estructura conceptual, una serie de prácticas que ofrecen situaciones de enseñanza específicas, y un impacto en múltiples aspectos de la vida que es clave comprender. Para tomar decisiones curriculares al respecto, la UNESCO recomienda *articular la naturaleza, la importancia y la relevancia de la informática para la sociedad y la educación* (2019:38).

El problema del enfoque transversal que llevaron adelante varios países, ha sido reportado por la UNESCO: *La integración de Ciencias de la Computación en otras asignaturas ha sido ineficaz. A pesar del enorme crecimiento del número de computadoras en las escuelas y del acceso generalizado a Internet, un estudio internacional sobre las políticas y las prácticas de uso de las TIC en los planes de estudio de 37 países (Plomp et al., 2009) ha llegado a la conclusión de que la integración del uso de las TIC en otras asignaturas escolares, como las ciencias y las matemáticas, por no hablar de la enseñanza de la informática, ha sido espasmódica y en muchas escuelas inexistente* (2019:41). El reporte recomienda identificar claramente los objetivos de aprendizaje, el modo de evaluación y la definición de unos estándares para la enseñanza de las Ciencias de la Computación.

Aun así, las iniciativas que intentan resolver la cuestión de cómo introducir los saberes de las CC apelando a un abordaje transversal, no debieran ignorar que en ningún caso este abordaje anula la necesidad de contar con la perspectiva disciplinar de las CC. Dado que las materias continúan siendo las organizadoras centrales del conocimiento escolar (Dussel, 2020), para incorporar la dimensión que aportan las Ciencias de la Computación, es necesario que algún espacio curricular aborde la computación como objeto de estudio.

Las propuestas de considerar a la Alfabetización digital como una competencia del siglo XXI, tampoco se traducen en la ausencia de las disciplinas, ya que el desarrollo de este tipo de competencia necesariamente se apoya en los conocimientos disciplinares (Rogiers, 2000).

Si acordamos en la actual concepción de la alfabetización digital y atendemos al planteo que la literatura especializada precisa acerca del enfoque transversal, **la perspectiva de las Ciencias de la Computación, no será la única disciplina responsable de su abordaje, pero sí una indispensable.**

De la misma manera que la enseñanza de conceptos claves de la biología o la historia son necesarios para abordar la problemática de la educación sexual integral, es necesario un espacio curricular que aporte los conceptos clave necesarios para alfabetizarse digitalmente.

## Referencias bibliográficas

Albarello, F. y Hafner Táboas, A. (2019). Programación y robótica: cómo y para qué. Análisis de las políticas educativas en Argentina. *Contratexto*, 2018(032), pp. 71–93.

Bonello, M.B. y Schapachnik, F. (2020). Diez preguntas frecuentes (y urgentes) sobre pensamiento computacional. *Revista Virtualidad, Educación y Ciencia*, 20 (11), pp. 156-167, Córdoba: Universidad Nacional de Córdoba

Boix Mansilla, V. & Gardner, H. (2005). ¿Cuáles son las cualidades de la comprensión? La enseñanza para la comprensión. Vinculación entre la investigación y la práctica, Paidós, Buenos Aires

CFE (2018). Resolución N° 343/2018 "Núcleos de Aprendizaje Prioritarios para Educación Digital, Programación y Robótica". Anexo I Núcleos y Anexo II. Lineamientos para la implementación. Cuadernillo de difusión <https://www.educ.ar/sitios/educar/resources/150123/nap-de-educacion-digital-programacion-y-robotica/download>

Computer Science Teachers Association (2017). CSTA K-12 Computer Science Standards.. <http://www.csteachers.org/standards>

Daino, M. y Rojas, M. C. (2006). La enseñanza de las ciencias naturales: desde las fuentes a las propuestas curriculares. Cuadernos de la Facultad de Humanidades y Ciencias Sociales-Universidad Nacional de Jujuy, (30), pp. 85-106.

Dussel, I. (2020). The Shifting Boundaries of School Subjects in Contemporary Curriculum Reforms. *ZEITSCHRIFT FÜR PÄDAGOGIK*, 66(5), pp. 666-689.

Fraillon, J., y otros (2019). IEA International Computer and Information Literacy Study 2018 Assessment Framework. <https://doi.org/10.1007/978-3-030-19389-8>

Ferreyra, H. A. (2004). Educación media en la República Argentina: avances, retrocesos y conflictos en el diseño e implementación de políticas para el sector: el caso de la transformación en la Provincia de Córdoba: (1997-2001). Universidad Católica de Córdoba. [http://pa.bibdigital.ucc.edu.ar/1426/1/TD\\_Ferreyra.pdf](http://pa.bibdigital.ucc.edu.ar/1426/1/TD_Ferreyra.pdf)

K–12 Computer Science Framework. (2016). <http://www.k12cs.org>.

Gobierno de la Ciudad de Buenos Aires. Ministerio de Educación. (2015) Diseño curricular nueva escuela secundaria.

Gobierno de la Provincia del Neuquén , Consejo Provincial de Educación, Resolución N° 1463/18. Diseño Curricular. <https://www.neuquen.edu.ar/resolucion-146318-diseno-curricular/>

Gobierno De Santa Fe - Ministerio de Educación. (1998). La transversalidad , la escuela y su compromiso con la comunidad. Documento de trabajo.

Gutman, S. y Siede, I. (1995). Formación Ética y Ciudadana. Documento de trabajo N.º1. Actualización curricular. Ciudad de Buenos Aires.

Kysilka, M. L. (1998). Understanding integrated curriculum, *The Curriculum Journal*, 9:2, pp.197-209.

Maggio, M. (2018). *Habilidades del siglo XX : cuando el futuro es hoy*. Documento básico, XIII Foro Latinoamericano de Educación. Santillana.

Martínez, M. C. (2020). El pensamiento computacional. Análisis de una competencia clave. *Virtualidad, Educación y Ciencia* 11.20 (2020): 226-229. <https://revistas.unc.edu.ar/index.php/vesc/article/view/27461/29029>

Martinez, C. y otros (2021) *Hacia un espacio de Ciencias de la Computación en la educación obligatoria*, Documento de trabajo. Fundación Sadosky.

Martínez, C, Martínez López, P., Gómez, Borchardt, M (2022) *Hacia un currículum emancipador de las Ciencias de la Computación*. Artículo presentado y aprobado por la Revista Latinoamericana de Economía y Sociedad Digital. Publicación pendiente.

Martinez, C. y Borchardt, M. (2021). Enfoques y perspectivas didácticas globales en la enseñanza de la computación. Simposio Argentino de Educación en Informática (SAEI) en el marco de las Jornadas Argentinas de Informática Virtuales (JAIIO), 12 y 13 de octubre de 2021. *Anales de SAEI 2021* <https://50jaiio.sadio.org.ar/pdfs/saei/SAEI-08.pdf>

Martinez V (2022) *Inteligencia Artificial y ética: el caso DALL.E* <https://www.infobae.com/opinion/2022/05/11/inteligencia-artificial-y-etica-el-caso-dalle/>

Moreno, M. (2009) *Los temas transversales: una enseñanza mirando hacia adelante*. En Busquets, M.D, et.al (2009) *Los Temas Transversales*. Santillana.

OECD. (2021). *Curriculum (re)design: A series of thematic reports from the OECD Education 2030 project*. Overview Brochure (Vol. 53, Issue 9).

Pardo, J. (2021) *La imposición de integrar asignaturas en ámbitos*. *El diario de la Educación*. <https://eldiariodelaeducacion.com/2021/06/09/la-imposicion-de-integrar-asignaturas-en-ambitos/>

Perrenaud, P. (2006), *Construir competencias desde la escuela*. Ediciones Noreste, J. C. Sáez Editor.

Roegiers, X (2000) *Saberes, capacidades y competencias en la escuela: una búsqueda de sentido*. *Innovación educativa* N°10.

Schapachnik, F. (2021) *Programación, Robótica, Ciencias de la Computación*. Despejando la ensalada epistemológica que la Informática trajo a la Educación. Charla ofrecida en el IV Congreso Municipal de Educación, Secretaría de Educación de la Municipalidad de Córdoba. [https://youtu.be/XOVX28J\\_JzU](https://youtu.be/XOVX28J_JzU)

Terigi, F. (2008). Los cambios en el formato de la escuela secundaria argentina: por qué son necesarios, por qué son tan difíciles. *Propuesta Educativa*, 1(29), 63–71.

UNESCO (2019). *Coding, Programming and the Changing Curriculum for Computing in Schools Report of UNESCO/IFIP TC3 Meeting at OCCE – Wednesday 27th of June 2018, Linz, Austria*

Wing, J. M. (2011). *Research Notebook: Computational Thinking—What and Why*. *The link Magazine*, 6, 20-23. <https://people.cs.vt.edu/~kafura/CS6604/Papers/CT-What-And-Why.pdf>

### **Piche: evolución del robot educativo soberano**

Correa, Carlos Maximiliano / maximiliano.c.correa@gmail.com

Etcheverry, Carlos / etcheverry95@gmail.com

Etcheverry, Pablo Leonel / etcheverrypablol@gmail.com

Ferreira Szpiniak, Ariel / aferreira@exa.unrc.edu.ar

Ludueña, Paula Soledad / paula.ludueña@gmail.com

Universidad Nacional de Río Cuarto

#### **Resumen**

Motivados por el propósito de crear un robot basado en desarrollos locales, adaptado a nuestro medio y contexto social, que satisfaga las demandas educativas y lúdicas, y con el objetivo de abrir nuevos caminos hacia la robótica, desde 2019 y en el marco del Proyecto AMULEN, hemos trabajado en el desarrollo de la nueva versión del robot educativo soberano, al cual hemos llamado *Piche*. Su nombre proviene de la lengua *rankülche* que significa *peludo, armadillo*. Los *Rankülches*, más conocidos como *Ranqueles*, son un pueblo originario muy referenciado en la zona de Río Cuarto. A su vez, el peludo o *Piche* es uno de los animales autóctonos en éstas zonas geográficas, lo que nos inspiró a crear un robot basado en nuestras raíces; que ayude a descubrir vocaciones científicas en el área de programación y robótica y que contribuya a la soberanía tecnológica.

**Palabras clave:** Programación, Robótica, Educación, AMULEN, Piche.

#### **1. Introducción**

Debido a que hoy en día existen fuertes políticas públicas para reducir la brecha digital de 1º, 2º y 3º orden; que desde la creación del Programa Conectar Igualdad, se promueve la inclusión digital e impulsa la industria nacional y que se encuentran muchas dificultades al momento de disponer de un robot para aprender programación y robótica en las aulas, es preciso gestar el desarrollo de un robot que permita abarcar todas las dimensiones descritas, en prácticas concretas tanto en los espacios áulicos, así como también en la producción de material didáctico, en las formaciones y capacitaciones que garanticen el vínculo “conocimiento-sociedad” (en términos de inclusión socio-educativa), evitando de esta manera que la educación digital se limite a equipar escuelas, en especial con tecnologías que responden a otras realidades socio-culturales y podrían no resultar aptas para recrearlas o adaptarlas a nuestros contextos, como los dispositivos cerrados provenientes de empresas internacionales que proveen un mismo paquete tecnológico a todos los países que deciden adoptarlo, sumado a software propietario sin posibilidades de adaptación.

En síntesis, el desafío es crear un robot educativo que tiene como objetivo brindar una doble solución, por un lado dar respuesta a las demandas educativas y lúdicas actuales y por el otro, ofrecer acceso a la tecnología a todas las personas que quieran aprender o profundizar su conocimiento en robótica y programación.

#### **2. Antecedentes**

En el marco del Proyecto AMULEN desarrollamos un robot educativo. AMULEN es una palabra proveniente de la lengua *rankülche* y significa *caminar* o *progresar*. Los *Rankülches*, más conocidos como *Ranqueles*, son un pueblo originario del suelo que habitamos en el sur de Córdoba; La Pampa; sur de San Luis y oeste de Buenos Aires. *Rankül* significa *caña* o *carrizo*, y *che* quiere

decir *persona* o *gente*, es decir *gente de los cañaverales*. Su nombre y su lengua nos inspiraron en el desarrollo de un dispositivo adaptado a nuestro medio y contexto social, basado en raíces locales y soberanía tecnológica, que pudiera satisfacer las necesidades de los distintos niveles educativos.

El proyecto AMULEN fue propuesto y aceptado en la segunda convocatoria a Proyectos de Estímulo a la Vocación Emprendedora (2019-2020), y actualmente participa de la tercera (2021-2022). Además, forma parte de las acciones desarrolladas por el Proyecto de Investigación “Pensamiento Computacional y los Núcleos de Aprendizajes Prioritarios de Educación Digital, Programación y Robótica (NAPEDRyP) desde un paradigma inclusivo y con compromiso social. Aportes para la implementación de una propuesta educativa digital concreta en contexto de territorio” (2021-2022), todos aprobados y financiados por la Universidad Nacional de Río Cuarto.

En la actualidad existe, tanto a nivel nacional como internacional, una creciente demanda insatisfecha de profesionales en las áreas de las ciencias exactas e ingenierías. Por ese motivo se definió que los integrantes del proyecto AMULEN y becarios (estudiantes de las carreras de computación e ingeniería de la UNRC) trabajaran en el desarrollo de un robot siguiendo un proceso productivo vigente en las Software Factory del país. Más allá del robot en sí, se consideró sumamente importante que los estudiantes becarios incorporasen experiencias y fortalezas en habilidades blandas y duras.

Durante la primera etapa del proyecto, se diseñó y construyó el primer prototipo del robot educativo basado en Arduino, para dar los primeros pasos en el abordaje de conceptos vinculados a la programación y robótica, como así también para ser una adecuada plataforma educativa y de investigación para las escuelas. Luego se adaptó un entorno de programación por bloques libre y se desarrollaron nuevos bloques que permiten interactuar de manera muy simple con él (Correa et. at, 2021). Al mismo tiempo, durante esta primera etapa, con el objetivo de conocer mejor los productos disponibles y los desafíos afrontados por los mismos productos (similares en el sector) al momento de implementar robótica educativa; se identificaron y estudiaron diversas empresas destinadas a brindar hardware y software orientados a la robótica educativa, entre ellas: IT10, RobotGroup, LEGO, Robobloq, Mis Ladrillos y Rasti. De todas ellas se estudiaron las diferentes características de los productos que éstas ofrecen; también se realizaron encuestas a docentes y directivos de escuelas primarias y secundarias de Río Cuarto y la zona. Una vez obtenidos los datos, se procedió a agruparlos en 3 grandes categorías: capacitación, equipamiento y experiencia del cliente. A partir de allí se delinearón los objetivos básicos: desarrollar un robot educativo libre y basado en industrial nacional, de costo accesible, modular, robusto, con buena autonomía de uso, tecnología 3D y programable mediante bloques (Correa et. al., 2021). Así surgió la primera versión del robot, que fue bautizado con el mismo nombre del proyecto, AMULEN.

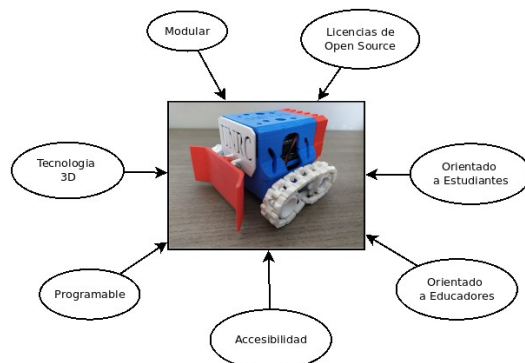


Figura 1: Características de AMULEN

### **3. Piche - Gestación**

A mediados del año 2020, equipos de investigación de la UNRC participaron de una reunión con el Parlamento Ranquel. El objetivo de la misma, fue presentar el Proyecto AMULEN y describir sus características: el desarrollo de un robot educativo soberano, libre, de bajo costo y desarrollado en nuestro país, para el abordaje de los NAPEDPyR y como juguete STREAM, desde un paradigma inclusivo, en contexto de territorio y con compromiso social. Previo a dicha reunión, y como parte de esa mirada desde la inclusión, el territorio y el compromiso social, AMULEN comenzó a interactuar con el Programa de Investigación “Historia y registros: frontera, etnicidad y racismo en el Cono Sur (siglos XVIII-XXI)”, también de la UNRC, para incorporar aspectos relacionados con la cultura ranquel.

Durante la reunión virtual, los miembros del Parlamento Ranquel manifestaron su beneplácito y compromiso para participar activamente del Proyecto AMULEN. Se conformó una agenda de trabajo con el objetivo de delinear algunas características del robot que le den identidad (por ejemplo su coraza, logo y colores) y en la elaboración conjunta de materiales de aprendizaje que, además del abordaje de los NAPEDPyR, también posibiliten trabajar aspectos relacionados con la identidad y la cultura ranquel, historia, geografía, flora y fauna autóctona. Podríamos decir que ese fue el puntapié inicial para la segunda versión del robot. Luego de compartir una serie de fotos y videos con el nuevo diseño del robot, desde el Parlamento Ranquel propusieron que se llamara *Piche*, dado que su aspecto y movimientos se asemejan a los de un peludo, que por otra parte es un animal autóctono muy querido por la Comunidad.

#### **3.1 Proceso Productivo - Metodología de Desarrollo**

Durante la primera etapa del Proyecto AMULEN, se contempló trabajar bajo la metodología SCRUM, logrando sentar las bases de la misma en el equipo, teniendo como pilar la presencialidad; por lo que en la segunda etapa y ya conociendo las ceremonias con los roles definidos, se decide (influenciados por la cuarentena Covid-19), implementar una especie de metodología híbrida entre SCRUM y Kanban, teniendo presente que el proceso de desarrollo de los diferentes módulos (inherentes al robot Piche) se fabrican de forma desacoplada y completamente remotos. A su vez maximizamos el uso de los procesos de revisión de código, mediado por Pair Programming de una manera necesaria para generar versiones de código betas o pre-alphas estables.

En la segunda se incorporaron nuevos integrantes al equipo por lo que fue necesario realizar una inducción del trabajo realizado hasta el momento. La misma, se realizó de una manera más rápida, con el objetivo de ir explicando y capacitando al nuevo miembro del equipo a medida que se iban desarrollando las tareas. Esta forma de trabajo, nos resultó más práctica y eficiente. Priorizando en todo momento los roles como las capacidades adquiridas de los integrantes mediados por capacitaciones en las herramientas de trabajo.

#### **3.2 Programación mediante Bloques**

Como parte de los trabajos a futuro planteados en la primera etapa, se propuso darle continuidad al desarrollo de nuevos casos de usos y mejorar los anteriores. A raíz de eso, realizamos dos grandes aportes: por un lado, se refactorizó el código de los bloques ya existentes para que sea escalable en relación a las nuevas funciones (reutilización de código existente) y mantenible, teniendo como principal pilar los principios SOLID como el mantener un bajo índice de acoplamiento incrementando la cohesión de los módulos desarrollados.

Por otro lado, se pensó como estrategia de modelado de las nuevas funcionalidades y guiados por las necesidad de que los diferentes bloques (funciones atómicas) deberían interactuar entre sí de manera independiente del contexto (escenario programado) a los siguientes bloques:

- Mover (Atrás / Adelante)  
Mediante el uso de parámetros específicos y desarrollados por nuestra cuenta, se redujo el número de bloques necesarios para contemplar los casos de usos existentes. Los bloques Mover Adelante y Mover Atrás ahora pueden ser reemplazados por Mover (Adelante / Atrás), que permite el ingreso como parámetro del sentido del movimiento.
- Buzzer (Encender / Apagar)  
Piche ahora cuenta con un buzzer con el que se puede realizar sonidos permitiendo una interacción más con el usuario.
- Led RGB  
Se agregó un módulo RGB al cual se puede setear una amplia gama de colores a través de tres parámetros. Con este bloque Piche podría entregar a modo de feedback ciertos estados como el estado de carga de las baterías.
- Medir distancia  
Se agregó un sensor de ultrasonido con el objetivo de que Piche pueda interactuar de manera más dinámica con el ambiente. Mediante este bloque puede medir la distancia de los objetos que tiene a su alrededor y en base a eso tomar decisiones.

Para ello fue necesario permitir el ingreso de parámetros entre un rango de valores específicos y ya predefinidos. Como es el caso del bloque Mover, que permite el ingreso del valor Adelante o Atrás (bloques que fueron necesarios desarrollar), brindando la posibilidad de determinar la dirección del movimiento de Piche y el bloque Buzzer que permite ingresar los valores Encender o Apagar para determinar el estado del buzzer.

En las siguientes imágenes se puede observar el desarrollo de los bloques en las 2 etapas.

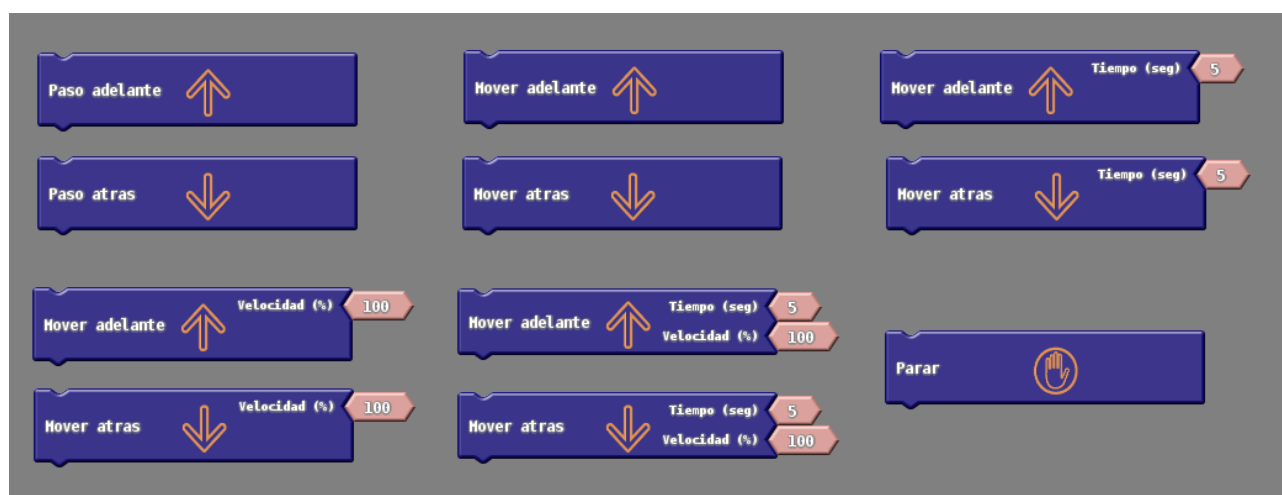


Figura 2: Bloques desarrollados en la Etapa 1.



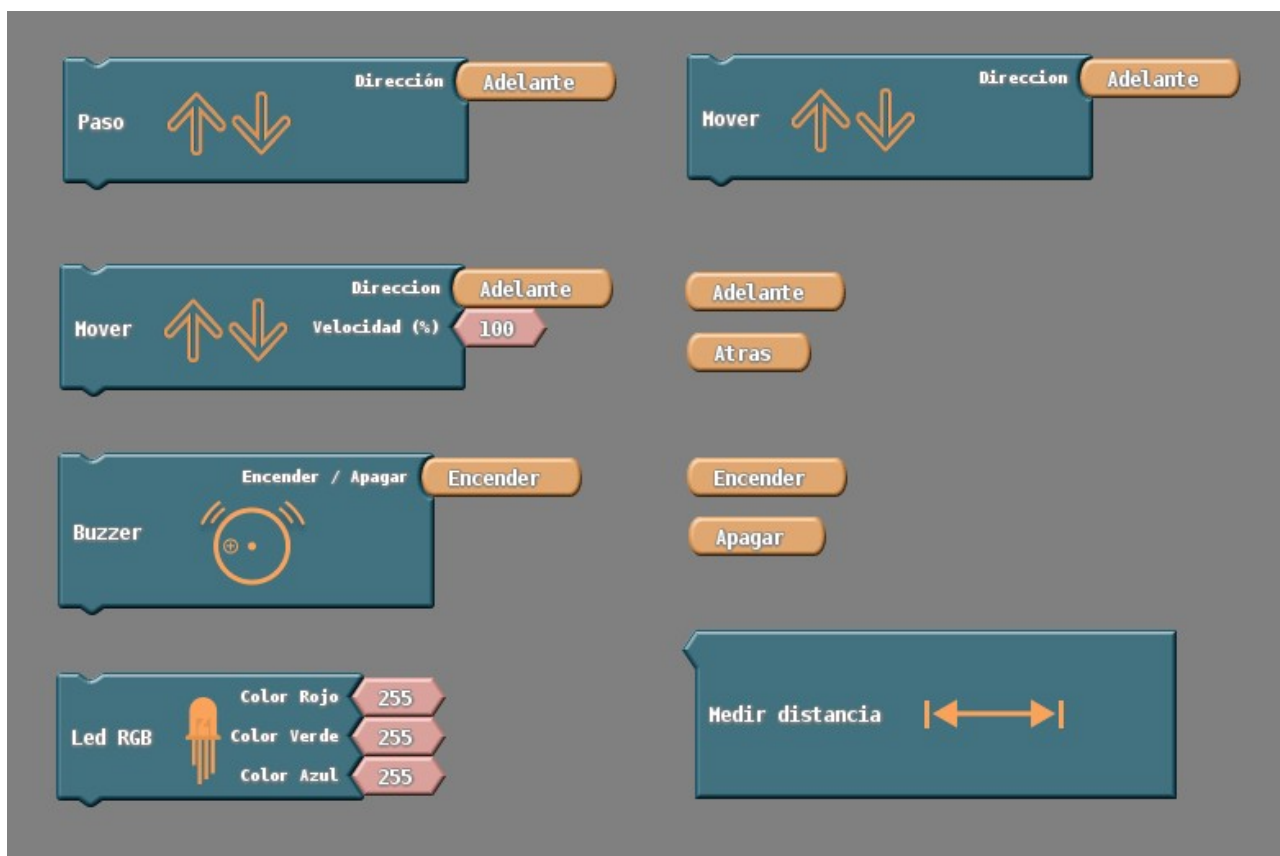


Figura 3: Bloques desarrollados en la Etapa 2.

Como se puede observar, para un mismo escenario podemos utilizar bloques que se desarrollaron en la etapa 1 o bloques que se desarrollaron en la etapa 2. Si tomamos como ejemplo el caso de dar un paso hacia adelante y uno hacia atrás, podemos usar los bloques ‘Paso adelante’ y ‘Paso atrás’ desarrollados en la etapa 1 o también lo podemos hacer con 2 bloques ‘Paso’, desarrollado en la etapa 2, insertando los parámetros ‘Adelante’ o ‘Atrás’. Dependiendo del nivel de abstracción, complejidad o nivel educativo en el que se aplique, se decidirá por un caso o el otro.

Para que sea más fácil distinguir el set de bloques a utilizar se tomó la decisión de utilizar colores diferentes.

### 3.3 Modelado y diseño de Piche

En esta segunda etapa se desarrolló una nueva versión con la finalidad de tener un diseño más robusto y cerrado. También se trabajó con archivos editables y paramétricos para futuras mejoras y desarrollos.

Piche está diseñado para soportar movimientos bruscos, golpes o caídas. Por ese motivo se desarrolló un modelo rígido, con la intención de que Piche no pueda ser deformado por medio de ninguna fuerza aplicada desde el exterior. De esta forma se busca durabilidad en la manipulación, tanto en el aula como en el hogar. Además, la carcasa posee encastres específicos para cada uno de los componentes electrónicos, como forma de minimizar las posibilidades de sufrir daños, fisuras de soldaduras o movimientos internos.

En la siguiente imagen se puede observar el nuevo diseño de carcasa realizado en esta segunda etapa. En flechas rojas se representan las fuerzas externas que piche puede soportar.

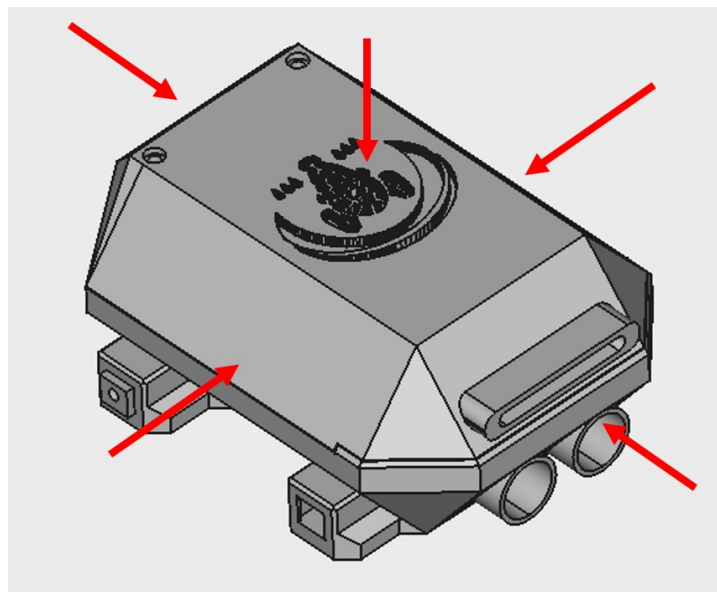


Figura 4: Nueva carcasa desarrollada en la Etapa 2.

Además, priorizamos la protección de la electrónica debido a su gran importancia, por ello se realizó un diseño de carcasa que fuera lo más cerrado posible, evitando así que puedan ingresar objetos extraños y pueda dañar la electrónica.

También, y atendiendo a la necesidad de trabajar con archivos editables, se decidió desarrollar a Piche en FreeCAD, software hecho principalmente para diseñar objetos de la vida real de cualquier tamaño. La elección de este software se basó principalmente en la motivación de tener, en todas las etapas de diseño, el código fuente, con la intención, de que nos permita acoplar al proceso de fabricación, el versionado de los diferentes pre-alphas o prototipos betas necesarios para generar tanto pruebas de concepto como pruebas de banco (simulación productiva).

En la siguiente imagen se muestran las mejoras hechas en esta nueva versión:

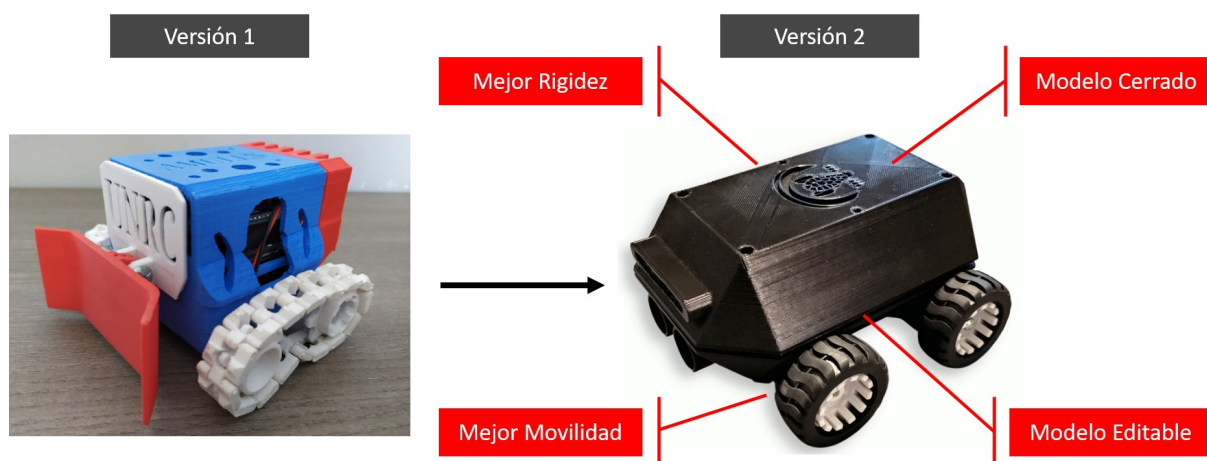
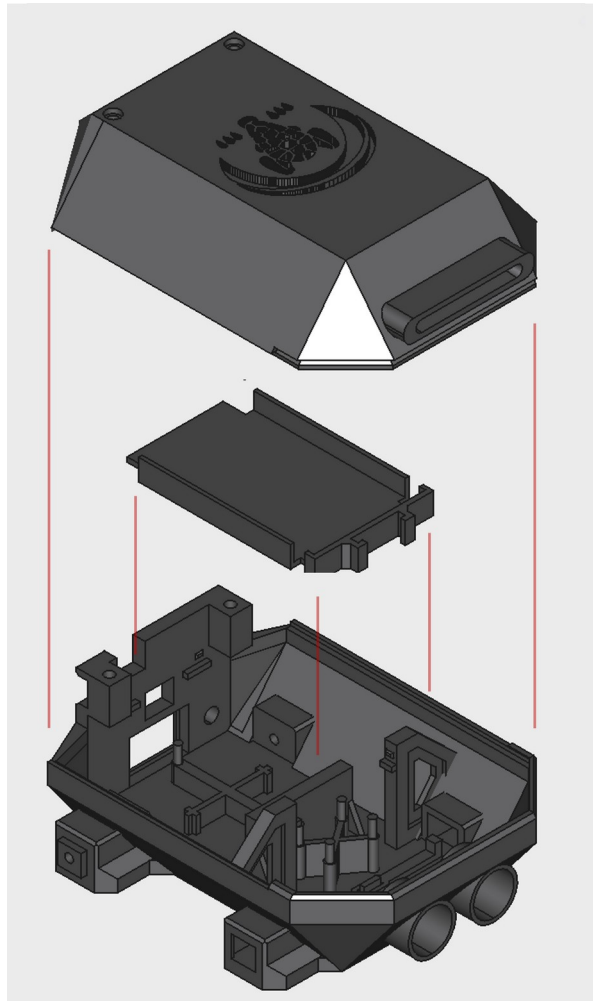


Figura 5: Mejoras entre la primera y la segunda versión.

Además, el diseño de los encastres de las carcasas y las ubicaciones de los diferentes componentes electrónicos están pensados para que, en caso de realizar reparaciones, sea de fácil acceso a los mismos, sin tener que desoldar las conexiones o forzar el cableado.

En la siguiente imagen se muestra el diseño interno de Piche:

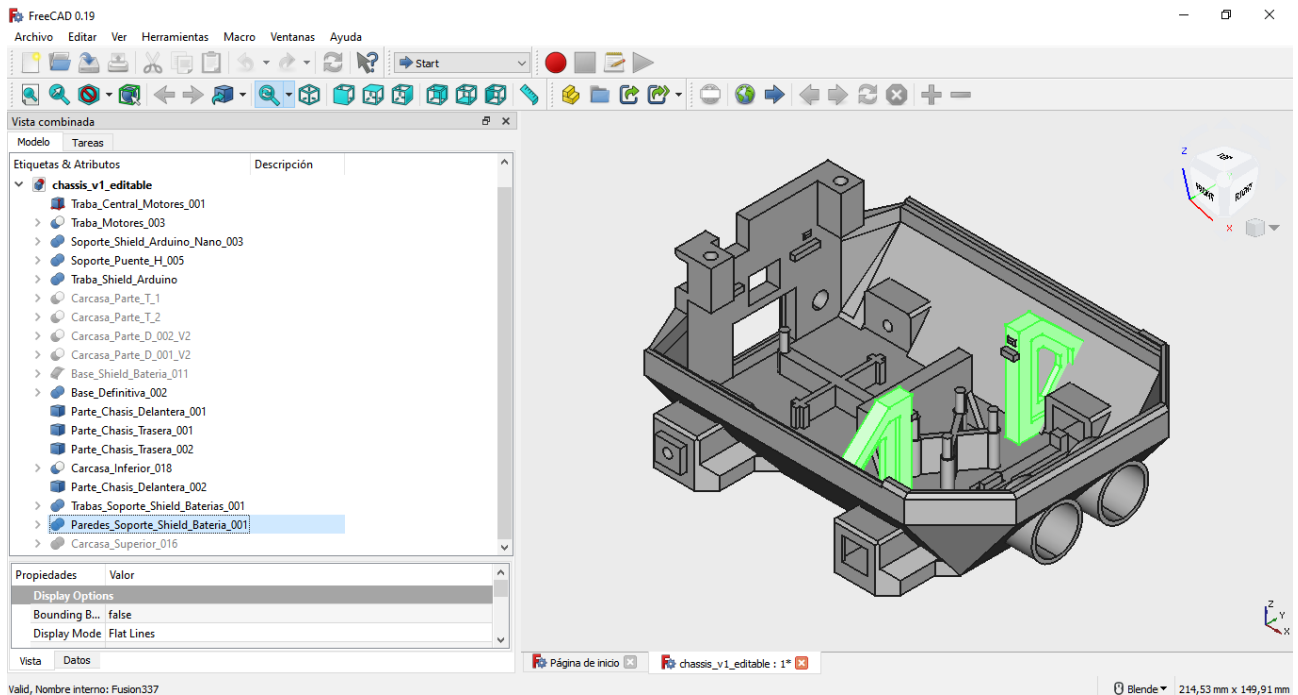


*Figura 6: Diseño interno de Piche.*

Como se puede ver en la figura anterior, son necesarias solo 3 piezas para imprimir a Piche. La carcasa inferior donde se montan los diferentes componentes electrónicos, un módulo intermedio de soporte para las baterías y fuentes necesarias para alimentar toda la electrónica y la carcasa superior. Las dimensiones y peso aproximado de piche luego de montados todos los componentes son:

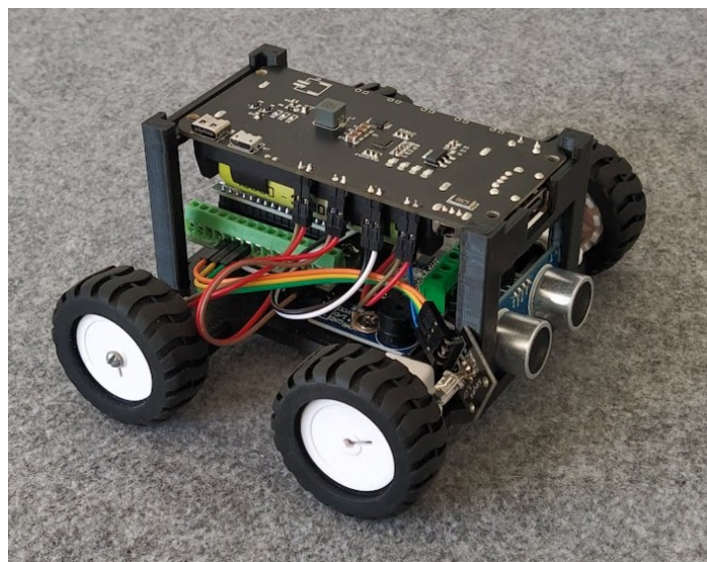
- Ancho = 14 cm
- Largo = 14 cm
- Alto = 8 cm
- Peso (kg) = 400 grs

Finalmente, para optimizar los tiempos de impresión de las carcasas, se creó un diseño específico que tiene como objetivo evitar los soportes (necesarios a la hora de realizar las impresiones de las diferentes piezas) o sobrantes al finalizar cada impresión.



*Figura 7: Diseño de la carcasa inferior de Piche en FreeCAD.*

Piche cuenta con una placa microcontroladora Arduino Nano, basada en el microchip ATmega328P (old bootloader), 4 ruedas todo terreno, 2 motores en las ruedas delanteras, 1 sensor ultrasónico en su parte frontal que le permite precisar distancias, actuadores de luz, sonido y piezas encastrables. Para su funcionamiento se optimizó su sistema de control de energía, haciendo uso de 1 batería níquel-cadmio de larga duración y una placa controladora que permite recargar la batería mediante micro USB.



*Figura 8: Componentes internos de Piche.*

### 3.4 Formación

Como el proyecto forma parte de los PEVE, los estudiantes participaron de los módulos del Trayecto en desarrollo emprendedor dictado por la UNRC para poder cumplir con los requisitos solicitados por la beca. Este trayecto tiene como objetivo general aportar formación integral a estudiantes y graduados en habilidades y competencias para emprender, fortaleciendo su proceso de formación y estimulando su vocación emprendedora. Para la aprobación del mismo se solicitaba la aprobación de los cuatro primeros módulos del trayecto en desarrollo emprendedor a los becados como complemento a las actividades comprometidas en el plan de trabajo de la beca EVE.

La temática abarcada de los cuatro primeros módulos del trayecto fueron:

1. Módulo N°1: Enfoque y Dimensiones del Emprendedorismo.
2. Módulo N°2: Oportunidades de Negocio.
3. Módulo N°3: Modelos de Negocios.
4. Módulo N°4: Design Thinking.

Como cierre del proyecto PEVE y del trayecto en desarrollo emprendedor, la UNRC invitó a todos a un encuentro llamado “Demo Day” con el objetivo de dar un cierre formal al proyecto. Previo al encuentro fué necesario crear un pitch del emprendimiento en formato digital, donde se potenciaban tanto las bondades y los objetivos de cada proyecto, como así también los aportes y desarrollos de los becarios. Los mismos se presentaron frente a representantes de la Incubadora de Empresas de la Universidad Nacional de Río Cuarto, miembros del Centro de Cultura Emprendedora (entidad que financia las Becas de Estímulo a la Vocación Emprendedora) y actores del ecosistema emprendedor (Córdoba Incuba y Agencia Córdoba Innovar y Emprender). Al finalizar se realizó una votación entre los participantes del “Demo Day”, resultando ganadora la presentación de PICHE, en categoría “Mejor pitch”.

### 4. Conclusiones y trabajo futuro

Como trabajo a futuro se plantea por un lado, realizar mejoras tanto en la funcionalidad como en el diseño del prototipo, de las cuales en relación al software y al hardware podemos mencionar:

- Nuevos casos de usos referidos a la gestión de movimiento, particularmente al movimiento rotacional.
- Refactorizar el diseño de Piche para que sea más pequeño, permitiendo encastrar módulos o elementos externos.
- Diseñar una placa electrónica propia, que permita reducir el tamaño de Piche, al igual que el peso y de esta manera repensar la lógica de conexionado de la electrónica.
- Presentar a Piche a la comunidad Ranquel, obtener retroalimentación y en base a eso trabajar en el look and feel externo para que se asemeje más a un armadillo.
- Coordinar el uso de Piche en dos escuelas, una de Río Cuarto y otra de Las Higueras, con el objetivo de evaluar la usabilidad.

Por otro lado, se plantea poder realizar el registro de la marca AMULEN para comenzar con la creación de un Spin Off universitario a raíz de este proyecto, con la ayuda de la Incubadora de Empresas de la Universidad Nacional de Río Cuarto. También tenemos como objetivo registrar la carcasa de Piche como modelo industrial, estimar el costo de su producción en serie y comenzar el diseño de materiales de aprendizaje (con desafíos y soluciones comentadas a nivel primario y secundario) para abordar los conceptos de pensamiento computacional y los NAPEDPyR desde un paradigma inclusivo y con compromiso social.

## 5. Bibliografía y referencias

### 5.1 Bibliografía

1. Astudillo, G. J.; Bast, S. G.; Willging, P.; Segovia, D. ; Castro, L. & Distel, J. M. (2019) *Estrategias innovadoras en los procesos de enseñanza y de aprendizajes de informática*. XXI Workshop de Investigadores en Ciencias de la Computación. WICC19 (pp.600-604) . San Juan. ISBN 978-987-3984-85-3
2. Benítez Larghi, S. (2020). *Desafíos de la inclusión digital en Argentina. Una mirada sobre el Programa Conectar Igualdad*. En Revista de Ciencias Sociales, DS-FCS, vol. 33, n.º 46, enero-junio 2020, pp. 131-154. Publicación del Departamento de Sociología de la Universidad de la República. Uruguay.
3. Bordignon, F., y Alejandro, I. (2015). *Diseño y construcción de objetos interactivos digitales*. Edit. UNIPE Editorial Universitaria. <https://libros.unlp.edu.ar/index.php/unlp/catalog/book/478>
4. Burbules, N. (2011). *Entrevista a Nicholas Burbules. Educación y tecnologías : las voces de los expertos. Conectar Igualdad*. <https://www.academia.edu/>
5. Compañ-Rosique, P., Satorre-Cuerda, R., Llorens-Largo, F., Molina-Carmona, R., (2015). *Enseñando a programar: un camino directo para desarrollar el pensamiento computacional*. RED- Revista de Educación a Distancia, 46 (11). <https://revistas.um.es/red/article/view/240191/182931>
6. Correa, C. M.; Etcheverry, P. L.; Ferreira Szpiniak, A. (2021). *AMULEN: robot educativo soberano*. Jornadas Argentinas de Didáctica de las Ciencias de la Computación.
7. Cucuzza, G. (2019). *Sobre los NAP de Educación Digital, Programación y Robótica: ¿Otra vez sopa?* <https://paraqueseapan.blogspot.com/2019/01/sobre-los-nap-de-educacion-digital.html>
8. Ferreira Szpiniak, A; Etcheverry, P. (2019). *Herramientas para descubrir vocaciones científicas y vincular la universidad con la escuela secundaria: talleres, olimpiadas y festivales de robótica y programación*. Segundas JADiPro. Córdoba. 7 y 8 de junio 2019.
9. Ferreira Szpiniak, A; Locati, M. (2019) *¿Para qué la robótica en la escuela? Una mirada nacional situada en la Provincia de Córdoba*. Segundas JADiPro. Córdoba. 7 y 8 de junio 2019.
10. Maggio, M. (2012). *Enriquecer la enseñanza. Los ambientes con alta disposición tecnológica como oportunidad*. Buenos Aires. Editorial Paidós.
11. Monsalves González, S. (2011). *Estudio sobre la utilidad de la robótica educativa desde la perspectiva del docente*. Revista de Pedagogía, 32(90), 81-117. <https://www.redalyc.org/articulo.oa?id=65920055004>
12. Willging, P. A., Astudillo, G. J, Bast, S., Occelli, M., Castro, L., & Distel, J. (2017). *Educación con Tecnologías: la Robótica Educativa Aplicada para el Aprendizaje de la Programación*. Memorias WICC 2017, XIX Workshop de Investigadores en Ciencias de la Computación, RedUNCI, 1174-1178. (ISBN: 978-987-42-5143-5).
13. Wing, J. M. (2006). *Computational Thinking. Viewpoint*. Communications of the ACM, Vol. 49, 33-35. <https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf>
14. Wing, J. M. (2011). *Computational Thinking: What and Why?* The Link Magazine, Spring.
15. Zabala, G. (2012). *Robots o el sueño eterno de las máquinas inteligentes*. ISBN 978-987-629-222-1. <http://www.sigloxxieditores.com.ar>

## 5.2 Referencias

16. ANSES. Observatorio de la Seguridad Social (2021). *Conectar Igualdad a 11 años de su creación. La consagración del derecho a enseñar y aprender*. Dirección Estudios de la Seguridad Social. Recuperado el 20 de junio de 2022 de: <http://observatorio.anses.gob.ar/archivos/documentos/Conectar%20Igualdad%20a%2011%20an%CC%83os.pdf>
17. Núcleo de Aprendizajes Prioritarios de Educación Digital, Programación y Robótica. Educ.ar (2018). Recuperado el 20 de junio de 2022 de: <https://www.educ.ar/recursos/150123/nap-de-educacion-digital-programacion-y-robotica?from=86>
18. *Orientaciones pedagógicas de Educación Digital*. Ministerio de Educación de la Nación (2017). Recuperado el 20 de junio de 2022 de: <https://www.educ.ar/recursos/132262/orientaciones-pedagogicas-de-educacion-digital>
19. *Programación y robótica: objetivos de aprendizaje para la educación obligatoria*. Ministerio de Educación, Cultura, Ciencia y Tecnología de la Nación (2017). Recuperado el 20 de junio de 2022 de: <https://www.educ.ar/recursos/132339/programacion-y-robotica-objetivos-de-aprendizaje-para-la-educacion-obligatoria>
20. *Resolución CFCyE N° 225/04* (2004). Recuperado el 20 de junio de 2022 de: <https://cfe.educacion.gob.ar/resoluciones/res04/225-04.pdf>
21. *Resolución Consejo Federal de Educación N° 263/15* (2015). Recuperado el 20 de junio de 2022 de: <https://cfe.educacion.gob.ar/resoluciones/res15/263-15.pdf>
22. *Resolución Consejo Federal de Educación N° 343/18* (2018). Recuperado el 20 de junio de 2022 de: [https://www.argentina.gob.ar/sites/default/files/res\\_cfe\\_343\\_18\\_0.pdf](https://www.argentina.gob.ar/sites/default/files/res_cfe_343_18_0.pdf)
23. UNESCO. SITEAL (2020). *Sistematización de respuestas de los sistemas educativos de América Latina a la crisis de la COVID-19*. Recuperado el 20 de junio de 2022 de: [https://siteal.iiep.unesco.org/respuestas\\_educativas\\_covid\\_19](https://siteal.iiep.unesco.org/respuestas_educativas_covid_19)



## LA ESCUELA DE ROBÓTICA DE MISIONES COMO MODELO DE IMPLEMENTACIÓN DE METODOLOGÍAS DISRUPTIVAS DE ENSEÑANZA Y APRENDIZAJE

Ing. Solange Schelske - [solange.schelske@gmail.com](mailto:solange.schelske@gmail.com)

Lic. Sebastian Maximiliano Rossi - [maximiliano001@gmail.com](mailto:maximiliano001@gmail.com)

Prof. Marcelo David Benítez - [marceloinformatica12@gmail.com](mailto:marceloinformatica12@gmail.com)

### RESUMEN

La escuela de Robótica es un espacio educativo de gestión estatal no arancelado que ofrece una propuesta pedagógica en torno a la ciencia y a la tecnología, orientadas a la programación y la robótica educativa en el marco de las transformaciones culturales del Siglo XXI. La firma de un acuerdo de estas características representa una experiencia piloto, pionera en el país, en lo que refiere a implementación de un modelo pedagógico innovador, disruptivo y con equipamiento único en Latinoamérica. Está pensada para niños a partir de los 5 años y jóvenes que se fortalecerán como ciudadanos con la formación en robótica y programación. Es un espacio ideado para el desarrollo, formación y expansión de vocaciones tecnológicas con 6 trayectos con 4 ejes de trabajo diseño, construcción, electrónica y programación, implementándose en los 77 municipios de la provincia de Misiones conformando la Red de Espacios Maker de Robótica con más de 15.000 estudiantes cursando de manera virtual y presencial en el desarrollo de proyectos y experiencias mostrando el éxito en la implementación de las diferentes metodologías disruptivas de enseñanza y aprendizaje guiados por más de 450 facilitadores técnicos y pedagógicos.

**Palabras clave:** Robótica-Aprendizaje Activo-Flipped Learning-ABP

### 1. INTRODUCCIÓN

En el año 2016 nace la "Escuela de Robótica de Misiones" como proyecto de la Unidad de Gestión en TIC por iniciativa de la Presidencia de la Cámara de Representantes de Misiones a cargo del Ing. Carlos Rovira. En ese año se firmó un convenio marco de colaboración público-privado con las empresas Hewlett Packard e Intel donde se establecía la provisión de equipamiento único en latinoamérica para llevar adelante la propuesta pedagógica innovadora. En el año 2017 iniciaron los Trayectos "Aprendiendo con Robótica" con cuatro trayectos Tecnokids, Maker Juniors, TeensMaker y Team Inn, para estudiantes de 5 a 20 años y una línea para trabajar externamente con equipos de estudiantes de escuelas de todos los niveles. La Escuela de Robótica de Misiones es una experiencia superadora para iniciarse en el mundo del desarrollo de las competencias y habilidades del siglo XXI. En ella los participantes podrán descubrir cómo construir un robot, crear un videojuego, aprender a desarrollar aplicaciones para móviles, descubrir el mundo de la Inteligencia Artificial, inventar con el diseño 3D y familiarizarse con los sistemas electromecánicos y sus aplicaciones a la ingeniería. En el año 2019 se organiza la Secretaría de la Unidad de Gestión en TIC con tres direcciones, en la cual la Dirección de Diseño y Desarrollo Tecnológico posee un departamento exclusivo para ejecutar las funciones del Proyecto Escuela de Robótica de Misiones. Se amplía el rango de edades, desde los 5 años en adelante, sin límite de edad, y suma prácticas profesionalizantes para estudiantes mayores de 18 años que hayan transitado todos los niveles de su



trayecto. La Escuela de Robótica de Misiones, tiene 6 trayectos de formación denominados “Aprendiendo con Robótica” aprobados por el Ministerio de Educación de la provincia de Misiones. Que se desarrollan de lunes a sábados en tres turnos diferentes (Mañana, Tarde y Noche), los estudiantes asisten en contra turno a sus escuelas y realizan los talleres entre 1 hora y media, a 2 horas y media, dependiendo del trayecto.

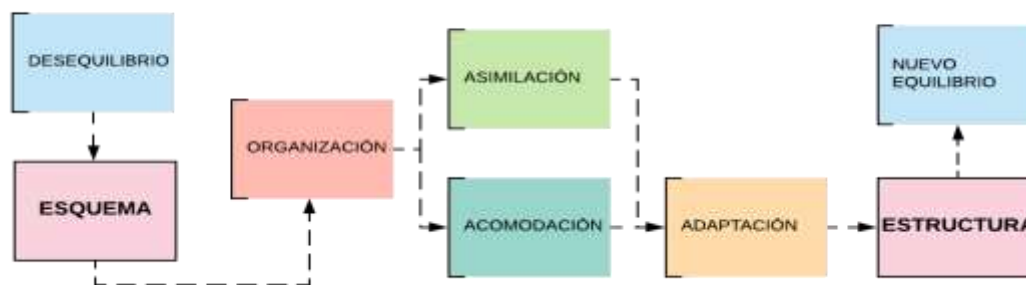


**Fig. 1:** Se pueden observar los trabajos de diferentes metodologías activas de aprendizaje en los seis trayectos que presenta la Escuela de Robótica desde los 5 años sin límite de edad máxima.

La Escuela de Robótica promueve los valores de emprendimiento entre los más jóvenes mediante el trabajo en equipo, la innovación y la creatividad. También se ofrece un acompañamiento para aquellos ciudadanos que se encuentran realizando proyectos y necesitan asesoramiento y/o orientación tecnológica. Las prácticas pedagógicas de la **Escuela de Robótica de Misiones** se enmarcan en la teoría constructivista del aprendizaje la cual sostiene que el conocimiento no se descubre, se construye: el estudiante construye su conocimiento a partir de su propia forma de ser, pensar e interpretar la información, desde una constante práctica de un proceso de experiencias y reflexión. El estudiante es quien participa activamente en su proceso de aprendizaje. Entre los principales referentes de la teoría constructivista se encuentran Jean Piaget, Vygotsky, Ausubel y Bruner.

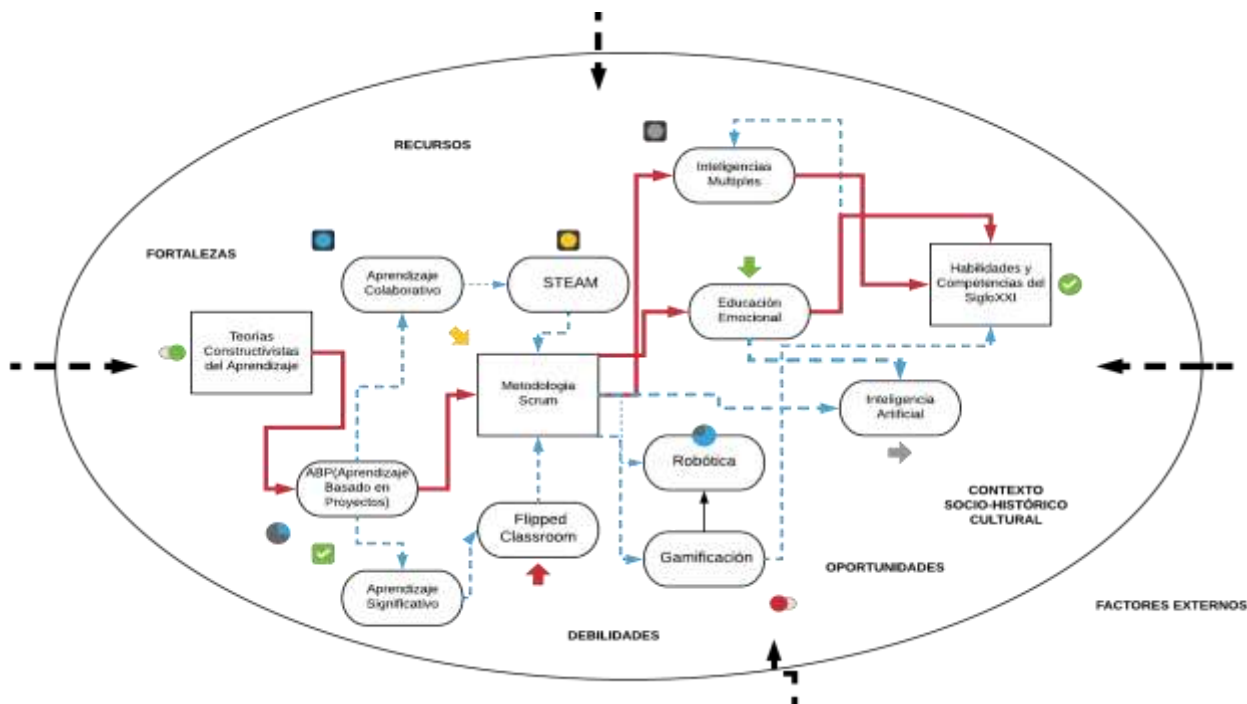
## 2. MARCO TEÓRICO

Piaget concibe el aprendizaje como un proceso interno de construcción, donde el estudiante participa activamente construyendo estructuras mentales cada vez más complejas. Bruner postula en sus producciones que el aprendizaje es un proceso donde los estudiantes construyen nuevas ideas o conceptos valiéndose de sus conocimientos y experiencias previas. El estudiante selecciona y transforma información, construye hipótesis, y toma decisiones, fiándose de sus estructuras cognitivas. El estudiante significa conocimientos, reconoce las similitudes o diferencias, diferencia y clasifica. La teoría Constructivista orienta el desarrollo de procesos de enseñanza aprendizaje desde la experiencia, donde se recomienda menos mensajes verbales del facilitador (mediador) y mayor actividad del estudiante.



**Fig. 2:** Modelo piagetiano en el que resalta que una de las ideas nucleares es el concepto de inteligencia como proceso de naturaleza biológica, donde la adaptación está siempre presente a través de dos elementos básicos: la asimilación y la acomodación generando un nuevo equilibrio mediante el desarrollo cognitivo.

El modelo Constructivista del aprendizaje implica el reconocimiento de que cada persona aprende de diversas maneras, requiriendo estrategias metodológicas pertinentes que estimulen potencialidades y recursos. Para el abordaje de los procesos de aprendizaje elegimos la metodología ABP (Aprendizaje Basado en Proyectos); es una metodología centrada en el estudiante como protagonista de su propio aprendizaje. El Aprendizaje basado en Proyectos permite organizar actividades en torno a un objetivo común, definido por los intereses de los estudiantes; Se fomenta la creatividad, la responsabilidad individual, el trabajo colaborativo, la capacidad crítica, la toma de decisiones, la eficiencia y la facilidad de expresar sus opiniones personales; Se busca dar respuesta a una problemática social, lo cual fortalece los valores y compromiso del estudiante con su contexto socio histórico cultural. Estamos de acuerdo con Gardner quien postulaba que el objetivo de la educación es desarrollar todas las inteligencias en todos los estudiantes y ayudar a las personas a alcanzar metas que sean apropiadas para la inteligencia que tengan más desarrollada. El incluir las Inteligencias Múltiples en los trayectos permite el desarrollo integral de las personas, contribuye a alcanzar un alto nivel de motivación y a lograr que los estudiantes sean más competentes y comprometidos con la sociedad. Por ello adoptamos la estrategia metodológica del Scrum. Dicha estrategia metodológica es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos donde cada integrante cumple una función específica acorde a sus competencias y saberes. En los procesos de aprendizaje se utilizará la Gamificación que es una técnica de aprendizaje que traslada la mecánica de los juegos al ámbito educativo-profesional con el fin de conseguir mejores resultados, ya sea para mejorar alguna habilidad, o bien evaluar el desempeño de los estudiantes, entre otros muchos objetivos. Al introducir la educación emocional como contenido transversal a todo el proceso de aprendizaje se busca reducir el impacto de problemas tan graves como el abandono escolar, el bullying, la apatía o la depresión entre los estudiantes. Los estudiantes deben superar adversidades, una correcta educación emocional facilita que los estudiantes puedan gestionar su propio proceso de aprendizaje, así se forman para desarrollar proyectos a largo plazo. Los estudiantes podrán controlarse en situaciones de estrés cognitivo y así gestionar el tiempo y los recursos de manera eficiente y eficaz. En el corriente año se busca implementar Flipped Classroom en los procesos de enseñanza y aprendizaje porque es un modelo pedagógico que transfiere el trabajo de determinados procesos de aprendizaje fuera del aula y utiliza el tiempo de clase, junto con la experiencia del docente, para facilitar y potenciar otros procesos de adquisición y práctica de conocimientos dentro del aula. Logrando así extender el aprendizaje y la participación de estudiantes y padres más allá de las horas de talleres.



**Fig. 3:** Representación esquemática de la educación disruptiva llevada adelante en los trayectos de la Escuela de Robótica de Misiones y la Red de Espacios Makers en el cual se puede visualizar las diferentes metodologías activas de aprendizajes entrelazadas por la educación emocional trabajando las habilidades y competencias para el siglo XXI teniendo en cuenta diversos factores internos y externos.

### 3. DESARROLLO DE LA EXPERIENCIA

**LOS MOMENTOS:** Es necesario preparar y planificar cada clase en toda su globalidad, no sólo los objetivos y la evaluación, sino también el uso del tiempo, del espacio, los recursos físicos que vamos a utilizar, cómo vamos a organizar los grupos, las actividades complementarias, etc.

Al inicio:	Durante la clase:	Al finalizar la clase:
<ol style="list-style-type: none"> <li>1) Posiciónate al lado de la puerta o entra de forma positiva.</li> <li>2) Saluda a los alumnos cordialmente.</li> <li>3) Sonríe.</li> <li>4) Habla sobre:                             <ul style="list-style-type: none"> <li>• sus últimas clases,</li> <li>• su fin de semana,</li> <li>• lo que echaban en televisión anoche,</li> <li>• lo felices que deben de estar en tu clase.</li> </ul> </li> <li>5) Usa un objeto o dibujo relacionado con el tema para una reflexión y conversación inicial.</li> </ol>	<ol style="list-style-type: none"> <li>1) Asegúrate de que los alumnos saben qué se espera.</li> <li>2) Da la clase para que dominen el tema en profundidad.</li> <li>3) Tiempo y ritmo.</li> <li>4) Minimiza el tiempo de transición.</li> <li>5) Variación. Evita la monotonía.</li> <li>6) Flexibilidad y adaptación.</li> <li>7) Material adicional.</li> <li>8) Feedback de los estudiantes.</li> </ol>	<p>Es importante generar una rutina para finalizar, para sintetizar y aprender de forma efectiva. Los estudiantes saben qué esperarse y están preparados para contribuir.</p> <p>Puede hacerse de forma individual, por parejas o en grupos.</p> <p>Podemos realizar un plenario que nos permita consolidar los principales objetivos de aprendizaje de la clase.</p> <p>Dar la oportunidad de evaluación informal.</p> <p>Hacer referencia de nuevo a objetivos de la clase.</p> <p>Puede ser 10 minutos o menos.</p>

**Fig. 4:** Esquema de implementación de ABP

Trayecto	TRENDKIDS	TECNOKIDS
Franja Etaria	De 5 a 6 años	De 7 a 8 años
Agrupación	Dos Integrantes	Tres/Cuatro Integrantes
Estrategia Metodológica	Aprendizaje Basado en Proyectos	

**Tabla 1:** Los dos primeros trayectos de la Escuela de Robótica

Trendkids es un espacio pensado para favorecer el desarrollo de las competencias tecnológicas, fomentar la búsqueda de información responsable, impulsando la creatividad digital, se propone trabajar mediante desafíos distintos para así impulsar la experimentación de diferentes estrategias de resolución de problemas. Se propone fomentar técnicas de aprestamiento y grafomotricidad para desarrollar en los alumnos la motricidad fina, en este sentido se proponen actividades diversas. Por otro lado, se plantean dos ejes transversales, el juego y la creatividad, teniendo en cuenta que este término se usa como sinónimo de muchos otros procesos, como originalidad, imaginación, pensamiento divergente. Pero a su vez, esto supone la generación de ideas que son al mismo tiempo originales y útiles, es decir, que tienen valor. Las ideas creativas deben representar algo nuevo y ser apropiadas y relevantes para la resolución de un problema determinado. En cuanto al juego, para Winnicott el jugar es un movimiento, un proceso que se está realizando, y del cual no importa el contenido, sino que importa en tanto capacidad de jugar, como testimonio de la creatividad de esa persona; es decir, en este espacio se promoverá espacios donde los alumnos puedan aprender robótica jugando desarrollando así su creatividad e imaginación. El Aprendizaje Basado en Proyectos Colaborativos (ABPC), empleando como estrategia didáctica, desarrolla en los estudiantes, motivación hacia la búsqueda y producción de conocimientos. La experiencia es conducida por el facilitador del curso; para recoger información, durante y al finalizar la experiencia, empleando las técnicas: observación, entrevista en profundidad y testimonios focalizados. Los estudiantes expresarán sus experiencias en cuanto al proceso de generación de conocimientos y sentimientos respecto a la actividad realizada. Describirán la experiencia adquirida al emprender proyectos colaborativos que buscan solucionar problemas. Del análisis de la información obtenida emergen categorías en cuyo contenido se aprecia la aplicación de ABPC. El modelo Constructivista del aprendizaje implica el reconocimiento de que cada persona aprende de diversas maneras, requiriendo estrategias metodológicas pertinentes que estimulen potencialidades y recursos. Para el abordaje de los procesos de aprendizaje elegimos la metodología ABP (Aprendizaje Basado en Proyectos); es una metodología centrada en el estudiante como protagonista de su propio aprendizaje. El Aprendizaje basado en Proyectos permite organizar actividades en torno a un objetivo común, definido por los intereses de los estudiantes; Se fomenta la creatividad, la responsabilidad individual, el trabajo colaborativo, la capacidad crítica, la toma de decisiones, la eficiencia y la facilidad de expresar sus opiniones personales; Se busca dar respuesta a una problemática social, lo cual fortalece los valores y compromiso del estudiante con su contexto socio histórico cultural. Para que los estudiantes logren la adquisición de conocimiento se emplea ABP (Aprendizaje Basado en Proyectos) adaptados según la franja etaria:



Fig. 5: Implementación del ABP para los seis trayectos de la Escuela de Robótica de acuerdo a las franjas etarias de los estudiantes comenzando por casos simples complejizándose en trayectos superiores.

Trayecto	<b>MAKER JUNIORS</b>
Franja Etaria	9 a 12 años
Agrupación	Micro Scrum 5 integrantes
Estrategia Metodológica	APRENDIZAJE BASADO EN PROBLEMAS

Tabla 2: Tercer trayecto de la Escuela de Robótica

La metodología del Aprendizaje Basado en Problemas es una innovación en la Educación, que se utiliza para la enseñanza de diversas áreas de conocimiento y, con frecuencia, para el trabajo de competencias profesionales determinantes en el perfil de estudiantes. El ABP intenta construir comunidades de aprendizajes colaborativos utilizando problemas reales. Se presenta la modalidad aprendizaje colaborativo en la aplicación de esta metodología. Al mismo tiempo, se estudia el importante papel del tutor o grupo facilitador en el Aprendizaje. Se estudian también los procesos evaluadores más adecuados para esta metodología, así como las ventajas y dificultades en la aplicación de este aprendizaje innovador en el ambiente educativo.

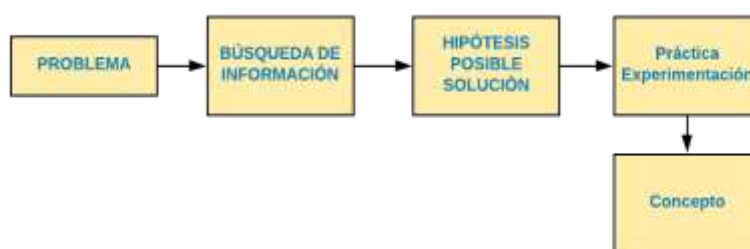


Fig. 6: Marco metodológico del Aprendizaje Basado en Problemas



Existen distintos modos de llevar a cabo las discusiones que despiertan el razonamiento. Según Torp y Sage (1998) los pasos a seguir son:

Presentar el problema
Identificar lo que sabemos, lo que nos hace falta saber y las ideas que tenemos al respecto. <i>(Preguntas abiertas (que no se limiten a una respuesta concreta); Preguntas ligadas a un aprendizaje previo (existe un contexto que sirve de preámbulo); Preguntas relacionadas con temas que despierten controversia y den lugar a variedad de opiniones.)</i>
Definir el problema al detalle
Reunir y compartir información
Generar posibles soluciones
6. Determinar el mejor abanico de soluciones. <i>(Hacer una lista de aquello que necesita hacerse para resolver los problemas.)</i>
Presentar la solución
Hacer una puesta en común

Trayecto	TEEN MAKER	TEAM INN	HIGH MAKER
Franja Etaria	13 a 15 años	16 a 18 años	19 años en adelante
Agrupación	SCRUM	SCRUM	SCRUM
Estrategia Metodológica	Aprendizaje Basado en Retos		

**Tabla 3:** Últimos trayectos a nivel etario de la Escuela de Robótica

El Aprendizaje Basado en Retos (ABR) es un enfoque pedagógico que involucra activamente al estudiante en una situación problemática real, significativa y relacionada con su entorno, lo que implica definir un reto e implementar para éste una solución. El Aprendizaje Basado en Retos tiene sus raíces en el Aprendizaje Vivencial, el cual tiene como principio fundamental que los estudiantes aprenden mejor cuando participan de forma activa en experiencias abiertas de aprendizaje, que

cuando participan de manera pasiva en actividades estructuradas. En este sentido, el Aprendizaje Vivencial ofrece oportunidades a los estudiantes de aplicar lo que aprenden en situaciones reales donde se enfrentan a problemas, descubren por ellos mismos, prueban soluciones e interactúan con otros estudiantes dentro de un determinado contexto (Cfr. Moore, 2013). El Aprendizaje Vivencial es un enfoque holístico integrador del aprendizaje, que combina la experiencia, la cognición y el comportamiento (Cfr. Akella, 2010). En el campo de la educación, destacados psicólogos y filósofos como John Dewey, Jean Piaget, William Kilpatrick, Carl Rogers y David Kolb han realizado importantes aportes a las teorías del aprendizaje a través de la experiencia. El modelo de Kolb (1984) describe al aprendizaje como el resultado integral de la forma en la que las personas perciben y procesan una experiencia.

## **SCRUM, UNA FORMA DE ORGANIZARSE PARA GESTIONAR**

**Proyectos con mayor agilidad:** La palabra SCRUM significa «melé», la jugada de rugby en la que todos los jugadores deben coordinarse y empujar al mismo tiempo para avanzar. Si no consiguen esta sincronización, la melé se cae. En la metodología ágil se utiliza esta expresión para crear un contexto de relación que permita que la organización de los proyectos sea lo más eficiente posible. ¿Cómo podemos utilizarlo en clase para la organización de proyectos?

Para un proyecto que dure, por ejemplo, 4 semanas vamos a organizar un SCRUM.

El equipo debe asignar el rol de Master SCRUM que va a ser el encargado de facilitar y garantizar que se cumple todo el proceso:

- Debe facilitar la comunicación entre todos los miembros.
- Garantizar que se explicitan todos los compromisos que se van adquiriendo.
- Hacer que se visibilicen todos los impedimentos que van surgiendo a medida que se desarrolla el proyecto.

### **LAS REUNIONES del SCRUM:**

En el transcurso del proyecto los equipos van a realizar tres tipos de reuniones:

**LA REUNIÓN DE INICIO DE PROYECTO**, en la que se acuerda cómo van a trabajar y cómo se van a repartir las tareas. Es un buen momento para construir su KANBAN que les va a servir de tablero-guía para todo el proceso y también como herramienta para hacer visibles los avances del equipo. Las **Daily SCRUM**, o reuniones diarias al inicio de clase, de pie (para conseguir más agilidad) y ante su KANBAN. Son encuentros que cada miembro del equipo responde a tres preguntas básicas:

- ¿Qué hiciste ayer en relación al proyecto?
- ¿En qué vas a estar trabajando hoy?
- ¿Qué problemas o impedimentos tienes?

**Estas reuniones no deben durar más de 5 minutos.**

Las **REUNIONES DE RETROSPECTIVA** son las que se realizan para valorar no solo la evolución de las tareas del proyecto sino también, y sobre todo, cómo evoluciona el equipo en cuanto a la relación y la comunicación y cómo se siente cada uno de sus miembros de forma individual.

## LAS PRESENTACIONES DE LOS TRABAJOS - SPRINT

Los sprint son presentaciones rápidas que van a hacer los distintos equipos de la evolución de sus proyectos. Dependiendo de la duración del proyecto podemos realizar uno o dos antes del SPRINT final de presentación de los proyectos.

Debemos animarlos a que realicen presentaciones impactantes utilizando el Q3 y la rueda de contenidos para conseguir resultados creativos.

Esta forma de trabajar aporta una enorme flexibilidad y capacidad de adaptación a los posibles cambios que se van dando en el transcurso de un proyecto.

En agilidad se utiliza la PROGRAMACIÓN EXTREMA (O PROGRAMACIÓN XP) como una forma de gestión que pone énfasis en la simplicidad, en la comunicación constante entre los miembros del equipo, en usar la retroalimentación con los clientes en el caso de las empresas, en ser valientes afrontando cambios, en respetar la opinión de todos y en ser humildes recurriendo a Baquellos que pueden enseñarnos todo lo que desconocemos. Nos parece interesante mencionar esta forma de trabajar porque puede ser muy interesante incorporar esta filosofía en el trabajo por proyectos en nuestras aulas. Además de estos valores la programación XP pone mucho hincapié en la planificación clara y simple y visible de todo el proceso, en utilizar pequeños prototipos que nos permitan anticipar los resultados del proyecto y probarlo antes de darlo por finalizado. Podemos incorporar aún otra buena práctica de este tipo de programación tan utilizada en los entornos de desarrollo de software. Los programadores deben siempre trabajar por parejas cuando están desarrollando un nuevo proyecto para garantizar un análisis constante de las ideas. Además de utilizar el SCRUM podemos incorporar algunas de estas ideas en la gestión de los proyectos de nuestros alumnos



PILA DE TAREAS	POR HACER	SE ESTÁ HACIENDO	TERMINADO
Tarea 1			
Tarea 5			
Tarea 4			



## VISIBILIDAD A TRAVÉS DEL KANBAN

Todo lo que el equipo hace, piensa, evoluciona y consigue debe reflejarse en un tablero (KANBAN) y organizarse a través de las distintas reuniones de los equipos.

Cada tablero debe llevar el nombre del equipo, y aunque aquí sugerimos un diseño, cada equipo puede personalizarlo (crear etiquetas con sus nombres, colocar una fotografía, un avatar...).

En la parte superior también debe aparecer el nombre del proyecto. En la casilla PILA DE TAREAS se colocan todas las tareas que deben realizarse para llevar a cabo el proyecto. En la columna POR HACER se escriben las tareas en pósits, con un símbolo o un color para indicar el alumno que las va a desarrollar. Los pósits se moverán a lo largo del tablero conforme avanza el trabajo de cada alumno durante las reuniones de reflexión (por ejemplo, al final de la semana). En la columna SE ESTÁ HACIENDO se indican las tareas que se están empezando a ejecutar. Los pósits permiten hacer visible el progreso. En cualquier momento podemos ver en qué punto estamos. El KANBAN es una estructura muy ágil, y si hay que incorporar una nueva tarea se añade un nuevo pósit. En la columna TERMINADO se recogen los pósits de las tareas que hayan sido terminadas. Los alumnos se sienten satisfechos cuando los pósits se acumulan en esta columna. En la parte inferior podemos añadir un espacio para anotar el resultado de las retrospectivas. Podemos también añadir una casilla de PARKING para colocar los pósits con una tarea a la que no saben dar continuidad o que consideran que necesita de la ayuda del profesor.

## DESIGN THINKING (DT)

Hemos visto a lo largo del libro herramientas y partículas que van a ayudarnos en la creación y gestión de proyectos de emprendimiento (y de cualquier otra índole) en nuestras aulas. La agilidad y muchos de los conceptos que se han expuesto hasta el momento son una adaptación de metodologías que se utilizan en los entornos empresariales. Unos entornos que buscan permanentemente la adaptabilidad a los cambios dado que la previsibilidad es cada vez más y más difícil.

El DT se ha erigido en los últimos tiempos como herramienta de gran utilidad enfocada a fomentar la innovación en las organizaciones. Es una metodología que se centra en comprender y entender a las personas beneficiarias de los productos o servicios y diseñar (de ahí su nombre) propuestas que satisfagan sus necesidades.

Lo más esencial del DT:

- Tener mucho conocimiento de los usuarios. El biopoema generacional que aparece al inicio del libro es un ejemplo sobre cómo conocer, por ejemplo, a nuestros alumnos.
- Se basa en la observación detallada del usuario.
- Imaginamos un «usuario tipo» para el que estamos diseñando nuestra propuesta.
- Generamos muchas ideas y propuestas.
- Construimos prototipos.
- Aprendemos a partir de las reacciones de los usuarios cuando interactúan con nuestro prototipo.

Cuando nuestros alumnos construyen sus proyectos debemos animarlos a que piensen bajo esta lógica. Que observen atentamente a los usuarios de sus proyectos, que construyan un usuario tipo,

que aprendan a buscar muchas ideas, que construyen prototipos antes de dar por finalizado sus proyectos y que aprendan de esos prototipos.

Los estaremos entrenando para vivir sin angustia la necesaria adaptación al cambio.

Elemento	Descripción
Idea general	Es un concepto amplio que puede ser explorado en múltiples formas, es atractivo, de importancia para los estudiantes y para la sociedad. Es un tópico con significancia global, por ejemplo, la biodiversidad, la salud, la guerra, la sostenibilidad, la democracia o la resiliencia.
Pregunta esencial	Por su diseño, la idea general posibilita la generación de una amplia variedad de preguntas. El proceso se va acotando hacia la pregunta esencial que refleja el interés de los estudiantes y las necesidades de la comunidad. Crea un enfoque más específico para la idea general y guía a los estudiantes hacia aspectos más manejables del concepto global.
Reto	Surge de la pregunta esencial, es articulado e implica a los estudiantes crear una solución específica que resultará en una acción concreta y significativa. El reto está enmarcado para abordar la idea general y las preguntas esenciales con acciones locales.
Actividades	Lecciones, simulaciones, recursos de contenido para desarrollar soluciones innovadoras, profundas y realistas.
Validación	Los estudiantes juzgan el éxito de su solución usando una variedad de métodos cualitativos y cuantitativos incluyendo encuestas, entrevistas y videos. El facilitador juega un rol vital en esta etapa.
Reflexión y diálogo	Mucho del aprendizaje profundo tiene lugar al considerar este proceso, se reflexiona sobre el aprendizaje propio, sobre las relaciones entre el contenido, los conceptos y la experiencia e interactuando con la gente.

**Tabla 4:** Marco metodológico del Aprendizaje Basado en Retos

## 4. RESULTADOS

### ESTUDIANTES CURSANDO PRESENCIAL 2021

INSCRIPTOS PRESENCIALES -> Cantidades Por Trayecto

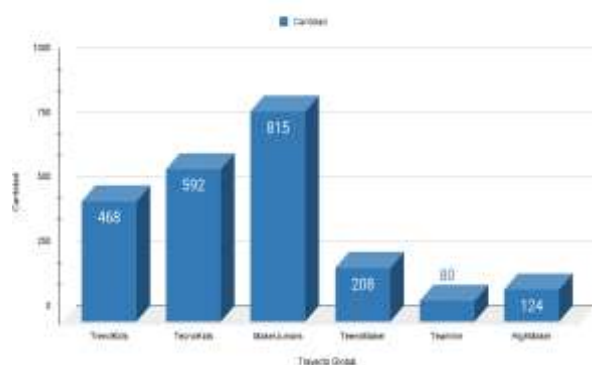


Fig. 7: Estudiantes totales cursando presencial por trayecto

INSCRIPTOS PRESENCIALES -> Cantidades Femenino y Masculino

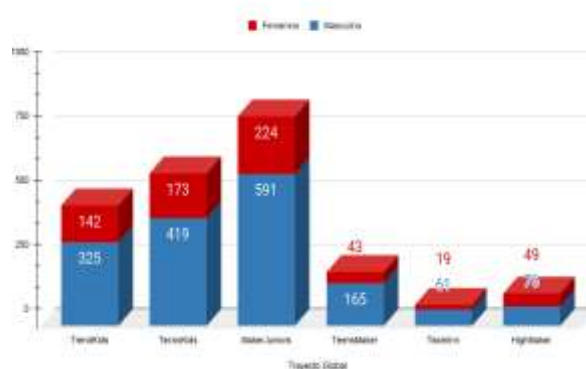


Fig. 8: Inscriptos presenciales por trayecto y género

## ESTUDIANTES CURSANDO VIRTUAL 2021

TRAYECTO	CANTIDAD	MASCULINO	FEMENINO
TRENDKIDS -5 y 6 años	1568	1001	567
TECNOKIDS -7 y 8 años	1992	1272	720
MAKERJUNIORS - 9 a 12 años	4487	2864	1623
TEENMAKERS -13 a 15 años	1470	938	532
TEAMINN -16 a 18 años	803	513	290
HIGHMAKERS - mayores de 18 años	2965	1893	1072
<b>TOTAL</b>	<b>13285</b>	<b>8480</b>	<b>4805</b>

**Tabla 5:** Total de estudiantes por trayecto en toda la red maker de la provincia cursando virtual

En la provincia de Misiones se establecieron 79 espacios Makers en todos los 77 municipios conformando una red de espacios maker donde se trabajan los seis trayectos de robótica categorizados por unidades etarias y se abordan transversalmente los 4 ejes de trabajo propuestos en todos los trayectos que son programación, electrónica, diseño y construcción. Desde el 2017 en su creación se han implementado a la fecha exitosamente diferentes metodologías activas de aprendizajes basadas en la ley provincial VI N°212 de Educación Disruptiva con más de 15 mil estudiantes distribuidos en toda la red maker cursando presencial y virtualmente. Las maneras de abordaje de las diferentes estrategias activas fueron mediante la incorporación de facilitadores pedagógicos y técnicos por trayectos con más de 450 en toda la red provincial. El acompañamiento a las Sedes Maker es una tarea sumamente importante que se lleva a cabo diariamente desde la Sede Central de la Escuela de Robótica. Para esto, se buscaron e implementaron diferentes herramientas o metodologías para poder acompañar y ayudar en el transcurso de las diferentes etapas del año lectivo en las sedes con el objetivo de ir fortaleciendo y mejorando los diferentes aspectos que hacen a la tarea tanto administrativa como pedagógica de los facilitadores. Una de esas acciones fue crear un espacio donde los facilitadores deben subir sus secuencias didácticas adaptadas al contexto en el cual se encuentran. Es sabido que cada sede presenta una realidad distinta a la de la sede central ubicada en la ciudad de Posadas. Estas diferencias están presentes en distintos aspectos como ser, en primer lugar, la cultura o costumbres propias de la localidad ya que cada una tiene fechas festivas particulares y/o valores diferentes, y la cantidad de estudiantes que asisten a los trayectos debido a la cantidad de habitantes que hay en cada localidad, y por otra parte, el acceso a los recursos materiales, ya sean los componentes necesarios para el desarrollo de las secuencias didácticas, como también la accesibilidad al internet y herramientas de comunicación como computadoras o celulares para mantener una comunicación continua y eficaz con las familias de los estudiantes.

Por esta razón, los facilitadores cuentan con un espacio donde cada vez que realizan una adecuación a la secuencia didáctica que se les brinda desde la sede Posadas, deben subirla con el objetivo de que podamos observar y evaluar dichas secuencias para realizar recomendaciones y/o correcciones a fin de que adapten las secuencias didácticas a su realidad de la mejor manera posible. Así también, tener la información de las adecuaciones que van realizando, nos brinda datos de cómo y por qué están trabajando de determinada manera, acercándonos a la realidad de cada sede estando preparados para resolver las dudas de la manera más eficiente posible cuando estas se presentan.

## 5. CONCLUSIONES

Los procesos de aprendizaje en la escuela de robótica propician la participación activa de los estudiantes tanto en forma individual potenciando la zona real de aprendizaje, como también el

trabajo entre pares. Los estudiantes desarrollan la creatividad día a día y junto con él, las ganas de descubrir, desarmar, armar y crear soluciones a problemas desarrollando una cultura maker para una sociedad empoderada en el saber hacer y no solamente consumista. Se observa que habilidades como el trabajo en equipo, resolución de problemas, diseño y construcción, la creatividad, pensamiento computacional y programación fueron incrementándose proporcionalmente con el transcurso de los meses. El registro diario y los distintos dispositivos de registro y evaluación nos fueron de gran ayuda al momento de reflexionar nuestras prácticas y encarar nuevas metodologías de enseñanza. Todos los proyectos llevados a cabo en cada trayecto de la Red Maker de la Escuela de Robótica visualizan un trabajo innovador en el cual los estudiantes se apropian del propio aprendizaje que es guiado por los facilitadores, y a su vez, les permiten participar de competencias zonales y provinciales exponiendo sus trabajos y retroalimentándose de otros proyectos del mismo trayecto en las diferentes sedes de la Red reflejando al mismo tiempo los diferentes modelos de implementación de las metodologías disruptivas de enseñanza y aprendizaje aquí abordadas.

## 6. BIBLIOGRAFÍA

- Ausubel, D. (1963). *Psicología del aprendizaje verbal significativo*, New York,
- Bruner, J & Haste, H, (1990) “La elaboración del sentido. La elaboración del mundo por el niño. Ediciones Paidós Ibérica, S.A. Barcelona.
- Bruner, J (1991) “Actos de significado”. Más allá de la revolución cognitiva.
- Bruner, J & Palacios, J (1995) “Desarrollo cognitivo y educación” Ediciones Morata, S. L Madrid.
- De Monterrey, O. de I. E. del T. (s/f). *Aprendizaje basado en retos*. Edu.co. Recuperado el 21 de junio de 2022, de <http://eduteka.icesi.edu.co/articulos/edutrends-10-2015>
- *Diez consejos para aplicar el aprendizaje colaborativo en el aula [Infografía]*. (s/f). Aulaplaneta.com. Recuperado el 21 de junio de 2022, de <http://www.aulaplaneta.com/2014/11/03/recursos-tic/diez-consejos-para-aplicar-el-aprendizaje-colaborativo-en-el-aula/>
- Gardner, Howard (1998). "A Reply to Perry D. Klein's 'Multiplying the problems of intelligence by eight'". *Canadian Journal of Education* 23 (1): 96–102. doi:10.2307/1585968. JSTOR 1585790.
- Institut. (2017, junio 16). *5 ejemplos de aprendizaje basado en problemas*. Mensalus. <https://mensalus.es/blog/infanto-juvenil/2017/06/ejemplos-aprendizaje-basado-problemas/>
- Piaget, J. La formación del símbolo en el niño. imitación, juego y sueño. Fondo de cultura económica de España.
- Piaget, J. La Construcción de lo real en el niño. Barcelona: Editorial Crítica, 1985.
- Inhelder, B & Piaget, J. De la lógica del niño a la lógica del adolescente. Buenos Aires: Paidós, 1972.
- Vergara, C. (2015, junio 3). *La teoría de los estilos de aprendizaje de Kolb*. Actualidad en Psicología. <https://www.actualidadenpsicologia.com/la-teoria-de-los-estilos-de-aprendizaje-de-kolb/>
- Vygotsky, L. S. (1962). *Pensamiento y Lenguaje*. Paidós.
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.
- Vygotsky, L. S. (1987). *El desarrollo de los procesos psicológicos superiores*. Austral.

## **Didáctica de la Programación en Educación Inicial: la co-construcción entre estudiantes y docentes**

Cecilia Haydée Exeni  
cexeni@iua.edu.ar  
Instituto Católico Superior

### **Resumen.**

La experiencia reseñada surge de las capacitaciones destinadas a docentes de Educación Inicial en enseñanza de programación. Las mismas fueron llevadas adelante por estudiantes del 2do año del Profesorado de Educación Inicial del Instituto Católico Superior, quienes enseñaron a maestrxs en ejercicio a utilizar las herramientas y recursos enviados por el Ministerio de Educación de la Nación para la implementación de los Núcleos de Aprendizajes Prioritarios de Educación digital, programación y robótica. Debido al contexto de pandemia, se realizó en modalidad virtual. Como resultado inmediato, lxs estudiantes capacitadorxs, tuvieron un acceso diferente a las prácticas y lxs maestrxs capacitadxs exploraron las herramientas, construyeron aprendizajes en programación y reformularon así sus propuestas educativas mediadas por tecnologías.

**Palabras clave:** programación, educación inicial, docentes, estudiantes

### **Introducción**

Este trabajo reseña el proyecto solidario implementado por el “Seminario TIC y la enseñanza en el Nivel Inicial”, del Profesorado de Educación Inicial del Instituto Católico Superior (INCASUP) durante los años 2021 y 2022. El mismo dio respuesta a la necesidad de capacitación de lxs docentes en servicio de diversas instituciones de Educación Inicial de la Provincia de Córdoba. Esta necesidad se origina en la presentación de los Núcleos de Aprendizajes Prioritarios (NAP) de Educación digital, programación y robótica durante el año 2017. A partir de ese momento las instituciones educativas de gestión pública de Argentina, recibieron herramientas y recursos para su implementación. En los propósitos de estos NAP, se manifiesta la necesidad de “incentivar a todos los chicos de nuestro país para que adquieran las habilidades que requiere el siglo XXI y formarlos para que sean capaces de entender y hacer un uso crítico de las tecnologías digitales en todos los aspectos de la vida y en el mundo del trabajo, en particular”. (Ministerio de Educación de la Nación Argentina, 2017, p. 3)

Si bien los NAP<sup>1</sup> definieron y secuenciaron los aprendizajes y contenidos desde la Educación Inicial a la Secundaria, lo cierto es que las tecnologías en la educación están presentes desde el 2006 en la

---

<sup>1</sup> Si bien hay NAP para todas las áreas, en este trabajo se hace referencia exclusivamente a los de Educación digital, programación y robótica.

Ley Nacional de Educación N° 26206, que fija la incorporación de las mismas en los procesos educativos, establece la producción de recursos educativos multimediales y multimodales, define los sitios WEB del Ministerio de Educación de la Nación (MEN) e incorpora la educación a distancia. Todo esto porque considera que el acceso a las tecnologías es un derecho y que son indispensables para la inclusión social. A respecto Silvia Bacher (2007) afirma:

“La sociedad de la información trae consigo una nueva concepción social, en la que los desconectados (sean niñas o niños en situación de calle, docentes que no se sienten seguros frente a sus alumnos o personas de la tercera edad que no acceden a las TIC) corren el riesgo de ser los nuevos segregados o más aún de convertirse en testigos carnales de una profundización aún mayor de exclusiones ya existentes. Hoy no es posible hablar de una brecha digital sino de brechas digitales enmarcadas en brechas sociales”. ( p. 1)

Las diferentes modalidades de educación dispuestas durante la pandemia posicionaron a las tecnologías en el centro de la escena y visibilizaron la profundidad de estas brechas, la urgente necesidad de acceso a la conectividad, a los dispositivos y a los conocimientos tecnológicos. A respecto Cecilia Martínez reflexiona: “Penosamente, de la brecha de saberes digitales no se habló. Distribuir saberes de computación hubiera permitido, (...) abordar algunos problemas locales a través de soluciones digitales. El acceso también permite la participación” (Otero, 2022,p.1)

En este marco, lxs docentes resignificaron de diferentes maneras los procesos de enseñanza y de aprendizaje. “Surgió el término “educación remota” que cada docente, cada escuela, cada equipo directivo, cada familia conceptualizó como pudo en función de los recursos económicos, humanos, tecnológicos y del contexto”. (Exeni, González, Melano, 2021). El whatsapp, las aulas virtuales, las herramientas para videoconferencia se incorporaron a los procesos educativos y adquirieron una asiduidad desconocida hasta ese momento. Sin embargo poco se avanzó en torno a las estrategias didácticas y a las potencialidades de las tecnologías para la construcción significativa de conocimientos.

Si bien los Jardines de Infantes que participaron de esta experiencia, contaban con los recursos dispuestos para el nivel (un Aula Digital Móvil con una notebook destinada a los docentes, entre 5 y 15 tablets en concordancia con la cantidad de estudiantes del Jardín, uno o dos robots, un proyector, una pizarra interactiva, un servidor fijo, un micrófono, un parlante y un servidor portátil o un disco rígido según el kit asignado. Todo esto, en un carro contenedor para guarda y traslado entre salas) no se utilizaron por el desconocimiento de docentes y directivxs, tanto de los dispositivos como de estrategias didácticas para la construcción de los conocimientos significativos que proponen los NAP

para Ed. Inicial.

Cabe destacar que es la primera vez que en Argentina se definen aprendizajes y contenidos para esta área del conocimiento y se tiene en cuenta la Ed. Inicial. Afirma Umaschi-Bers (2016)

“Es importante empezar en jardín; cuanto antes, mejor. Al ordenarle a la computadora lo que tiene que hacer moviendo bloquitos en un entorno gráfico, aprenden que cuando el programa no funciona se puede arreglar, pueden resolver el problema de manera creativa y volver a intentar. Es la nueva alfabetización, con competencias fundamentales para el siglo XXI, porque cuando uno programa también está pensando sobre su propio pensamiento.” (p1)

El desarrollo del pensamiento computacional y la enseñanza de la programación permiten imaginar un estudiante que se posicione como ciudadano con capacidad crítica y creador de herramientas y recursos tecnológicos. Mara Borchardt (2022) afirma:

“el uso es masivo, pero solo algunos se apropian de conceptos para diseñarlas, fabricarlas y comprenderlas, y, así, son capaces de participar como ciudadanos activos y opinar con fundamento sobre temas como la regulación de la inteligencia artificial, la neutralidad de la red o el voto electrónico” (Otero, 2022, p.1).

Para ello es menester incorporar estos aprendizajes “como objetos de enseñanza desde la Educación Inicial, a los fines de que contribuyan a desarrollar, construir y ampliar las posibilidades cognitivas, expresivas y sociales que los estudiantes ponen en juego y recrean cotidianamente en su encuentro con la cultura, enriqueciendo de este modo la experiencia personal y social en sentido amplio.” (MEN, 2017, p.6)

En la formación de docentes de Ed. Inicial se aborda el desarrollo del pensamiento computacional y la enseñanza de la programación desde el paradigma socioconstructivista con nociones teóricas y la exploración de recursos analógicos y tecnológicos. “El Pensamiento Computacional es una habilidad o competencia que se desarrolla con el tiempo (...) y en contexto, a través de experiencias educativas que requieren que los alumnos reconozcan la necesidad de ellas y las empleen según corresponda”. (Yadav & Berthelsen, 2021). Esto implica que los docentes debemos resignificar la enseñanza de las Ciencias de la Computación teniendo en cuenta el enfoque epistemológico, las notas identitarias de los sujetos, las potencialidades de las tecnologías en relación con los contextos y reformular la didáctica de la programación desde una perspectiva ética y política.

### **Descripción de la Experiencia**

### **De docentes devenidxs en estudiantes**

Para la implementación de los NAP, además de los dispositivos tecnológicos estaba prevista “la conectividad y la formación docente, que ayude tanto al desarrollo de las competencias de educación digital, como de las capacidades y saberes fundamentales. se equiparon a las escuelas con diferentes dispositivos tecnológicos.” (MEN, 2017). En concordancia, la Unidad de Educación Digital dependiente del Ministerio de Educación de la Pcia. de Cba organizó algunas pocas capacitaciones. Las mismas resultaron insuficientes tanto desde lo cuantitativo, debido a que lxs docentes carecen de los saberes propios de esta área de conocimiento, como desde lo cualitativo ya que lxs capacitadores desconsideraron las especificidades propias de la Ed. Inicial. Lo cierto es que llegado el año 2021, en muchos jardines de la Pcia de Cba, las herramientas digitales permanecían guardadas, sin ser utilizados ya que nadie sabía qué hacer con ellas.

Por otra parte, lxs estudiantes de Educación Superior que ingresaron en el 2020 debieron cursar su primer y segundo año de manera virtual debido al Aislamiento Social Preventivo y Obligatorio (ASPO). Casi todos los espacios curriculares resignificaron sus propuestas educativas en aulas virtuales, menos las del Campo de la Práctica Docente, ya que las instituciones permanecían cerradas.

Es preciso comentar que el INCASUP tiene en su Proyecto Educativo Institucional la formación solidaria. Lxs estudiantes asisten a las escuelas asociadas para construir los aprendizajes correspondientes a las Prácticas Docentes y además, relevan una necesidad de la comunidad. Luego diseñan e implementan un proyecto solidario que le dé respuesta. En general, estas instancias estuvieron asociadas a recolectar libros, juguetes, materiales didácticos, etc. Pero cuando en el 2021 se detectó como necesidad la falta de capacitación docente en torno a la programación y su didáctica, entonces se decidió diseñar un proyecto solidario interinstitucional, que permitiera a lxs estudiantes tener acceso (virtual) a los Jardines de Infantes, y a lxs maestrxs en actividad poder construir aprendizajes en el área de Programación y explorar las herramientas tecnológicas disponibles. Participaron directorxs y docentes de los jardines de gestión pública: Pablo Neruda, Gral. Mosconi, Merceditas de San Martín, María Elena Walsh, Bandera Argentina y la inspección de Villa María. Esta última convocó a unx maestrx de cada institución que luego socializaría los saberes construidos.

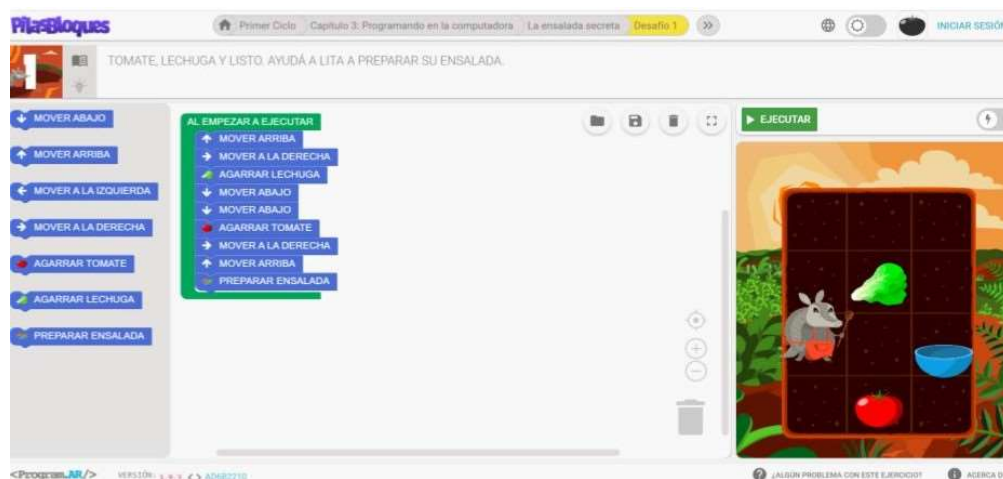
### **De estudiantes devenidas en docentes de programación**

En el Profesorado de Educación Inicial del INCASUP desde el año 2019, se agregó la enseñanza de la programación en el “Seminario TIC y su enseñanza en Educación Inicial”. A los fines del proyecto



las estudiantes de 2do año, construyeron conocimientos teóricos y prácticos en torno a la didáctica de la programación. Las clases se gestionaron en formato laboratorio y en concordancia, lxs estudiantes elaboraron hipótesis sobre el por qué y el cómo de la enseñanza de las ciencias de la computación en Educación Inicial. Las hipótesis daban cuenta de un imaginario en torno a la programación como de una aprendizaje muy complejo, asociado a la matemática, imposible de abordar en las infancias.

A los fines de comprobar las hipótesis experimentaron los desafíos de “Pilas Bloques” y se apropiaron de los NAP de Educación digital, programación y robótica poniendo el foco en las especificidades del nivel. “Pilas Bloques” es una aplicación para aprender a programar, desarrollada especialmente para el aula por Program.AR–Fundación Sadosky con la colaboración de Huayra. Se proponen desafíos con diversos niveles de dificultad para acercar a las y los estudiantes al mundo de la programación por medio de bloques.



Fuente: Pilas Bloque

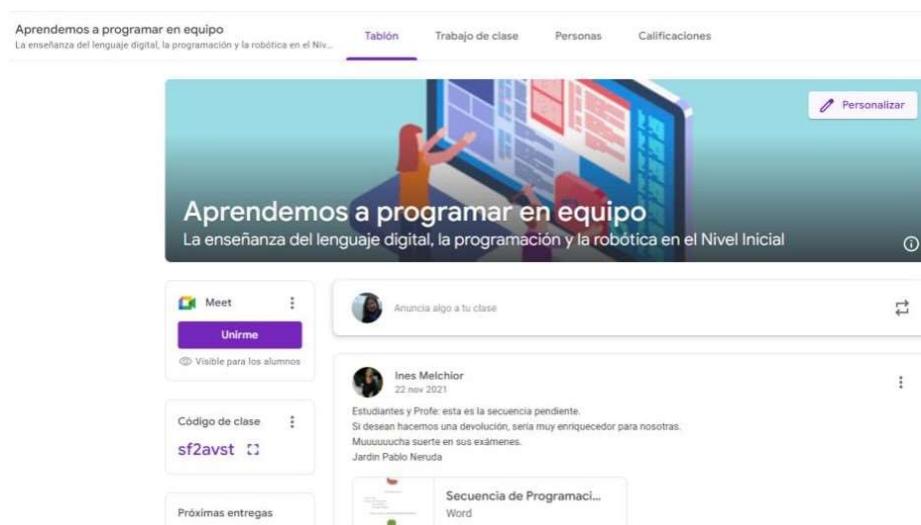
Estas instancias fueron decisivas ya que comprendieron las potencialidades tanto del recurso como de la programación para “desarrollar, construir y ampliar las posibilidades cognitivas, expresivas y sociales que los estudiantes ponen en juego y recrean cotidianamente en su encuentro con la cultura, enriqueciendo de este modo la experiencia personal y social en sentido amplio”. (MEN, 2017)

Luego exploraron libremente la plataforma “La hora del código” y jugaron con aquellos videojuegos destinados a “prelectores”, es decir para usuarios que aún no están alfabetizados. Finalmente aprendieron a utilizar Scratch Jr. que es la aplicación que traen las tablets para programar la “robotita”.

Paralelamente desde el espacio curricular “Didáctica General” planificaban una secuencia didáctica para enseñar programación a las maestras de Ed. Inicial. Como el ASPO se extendía, se decidió que la modalidad de esta capacitación sería virtual. Entonces por pares, abrieron un aula virtual de

Classroom en la que resignificaron la secuencia didáctica, considerando la educación mediada por tecnologías. Esto significó un nuevo desafío. Vale destacar que hasta aquí, nunca se había considerado la necesidad de enseñar a lxs estudiantes del Profesorado de Ed. Inicial, el uso pedagógico de aulas virtuales. La pandemia exigió repensar los procesos educativos mediados por tecnologías y considerar su incorporación aún en las primeras infancias. Esto implicó la resemantización del modelo socioconstructivista, la revisión de estrategias didácticas, la selección de herramientas y recursos que posibilitaran a lxs estudiantes la construcción de estos contenidos. Además abrió el debate en torno a las desigualdades en el acceso a dispositivos tecnológicos, la necesidad de conectividad, el cuidado de los datos, la ciudadanía y la soberanía digital.

De la misma manera que lo había aprendido ellxs, organizaron la enseñanza en los formatos laboratorio y taller con 3 instancias asincrónicas (en Classroom) de 4 hs en total y 2 sincrónicas (por meet) de 2 hs. cada una. La capacitación completa sumó 8 hs.



Fuente: aula virtual de estudiantes de INCASUP

En el aula virtual de Classroom para las instancias asincrónicas:

- Abrieron un foro para compartir saberes previos
- Compartieron los NAP y propusieron actividades para la lectura comprensiva y la resignificación de los mismos considerando los contextos y las notas identitarias de cada institución.
- Pusieron a disposición la Secuencia didáctica para Nivel Inicial disponible en la plataforma Program.ar
- Elaboraron un tutorial para facilitar la apropiación de Scrath Jr.
- Diseñaron una evaluación integradora

Organizaron dos clases sincrónicas:

-En la primera propusieron juegos en los que debías descomponer una acción cotidiana en partes, generalizar, abstraer y resolver una problema paso a paso. Luego buscaron que lxs capacitandxs asociaran el juego con los diferentes aspectos de la enseñanza de programación: algoritmos, repeticiones, variables, etc. Y que valoraran estos juegos a los fines de comenzar a desarrollar el pensamiento computacional en las infancias.

Luego guiaron la exploración de los desafíos de “Pilas Bloque” correspondientes al Primer ciclo. Y abrieron preguntas para resignificar los conceptos trabajados en la actividad anterior: algoritmos, repeticiones, variables y condicionales. Finalmente, en esta instancia sincrónica, solicitaron a lxs capacitandxs que propusieran aprendizajes y contenidos para articular con los juegos (analógicos y digitales) trabajados en la clase.

-En la segunda, recuperaron los saberes construidos en el encuentro anterior y experimentaron con Scrath Jr. Para esta instancia algunxs capacitandxs llevaron las tablets y la robotita disponible en los jardines. Así, a través de las pantallas lxs estudiantes capacitadorxs pudieron conocer estos dispositivos.



Fuente: Aprender conectados

## Resultados

Los aprendizajes logrados por lxs 38 docentes (entre docentes y directivos) de las 6 instituciones de Ed. Inicial que participaron, permitieron resignificar los procesos educativos considerando la incorporación de los NAP de Educación digital, programación y robótica y utilizar las herramientas disponibles en las instituciones. Además pudieron diseñar actividades articuladas con los aprendizajes y contenidos de los Diseños Curriculares y con el desarrollo de las capacidades fundamentales del Ministerio de Ed. de la Pcia de Cba: oralidad, lectura y escritura; pensamiento crítico y creativo; trabajo en equipo y resolución de problemas. Los desafíos de “Pilas Bloque” se articularon de la siguiente manera: “Dieta a base de churrascos” con el eje “Espacio” de matemática, “Las palabras de

Toto” con el Lenguaje escrito y “Coty empieza a dibujar” con el eje “Artes visuales” del área de Educación Artística.

Para lxs estudiantes, significó un acercamiento a los objetivos de las Prácticas docentes de 1er y 2do año del Profesorado que proponen:

“una aproximación al campo de intervención profesional docente y al conjunto de las tareas que en él se desarrollan. En tal sentido, compromete una doble intelección: a) la descripción, análisis e interpretación de la multidimensionalidad que caracteriza las prácticas docentes y de enseñanza y los contextos en que se inscriben, es decir, la orientada a la comprensión de las condiciones objetivas de producción; b) la que remite al sujeto de estas prácticas, la implicación de su subjetividad e identidad al incluirse desde una historia y una trayectoria singular que requiere procesos de objetivación mediados desde una posición de reflexividad crítica.” (Ministerio de Ed. de la Pcia. de Cba, 2011, p.17)

También les permitió reconocer las competencias tecnológicas que poseen y que les facilita la apropiación del pensamiento computacional y de los videojuegos disponibles para la enseñanza de programación. En los encuentros sincrónicos lxs docentes de los jardines solicitaban otros aprendizajes: cómo instalar una aplicación, abrir un aula virtual, descargar videos, etc.

Todxs coincidieron en la urgente necesidad de:

- que la conectividad sea un derecho al que debe tener acceso toda la comunidad educativa.
- la continuidad de capacitaciones en servicio
- de articulación con la Educación Primaria y Secundaria para un proyecto gradual que instale la enseñanza de la programación.
- servicio técnico que realice el mantenimiento de los dispositivos tecnológicos.

Esta experiencia se popularizó en dos medios masivos de comunicación provinciales: “La voz del interior” y Canal 10.

<http://www.pearltrees.com/ceciexe/instituto-catolico-superior/id15597811/item455373562>

<https://www.youtube.com/watch?v=2beE25I52BA>

Consecuentemente surgieron nuevas demandas por parte de docentes de Ed. Inicial y Primaria y la reedición de la experiencia durante el 2022. Además, lxs estudiantes que este año terminan y se

preparan para su Práctica final, solicitaron incorporar estos aprendizajes y propusieron secuencias didácticas basadas en la enseñanza de programación en las escuelas asociadas.



Estudiantes y docentes en el INCASUP aprendiendo a Programar

## Conclusiones

En tiempos de visibilización de brechas digitales, etarias y de conocimiento, esta capacitación habilitó el coprotagonismo en las aulas lo cual produjo repercusiones socioculturales, epistemológicas, pedagógicas y didácticas. Lxs estudiantes demostraron solvencia en el uso de las tecnologías y lxs docentes modificaron sus prácticas para co-construir juntxs nuevos aprendizajes. El encuentro permitió repensar las potencialidades de los procesos educativos con tecnologías y la necesidad de asumir la incorporación de los NAP de educación digital, programación y robótica por parte del docente, de las escuelas y de quienes las gestionan; superando miradas reduccionistas, tecnocráticas y simplistas y, sobre todo, dotando a lxs docentes con nuevas herramientas de acción y reflexión que permitan un uso crítico de las tecnologías en la educación.

En ese marco es menester continuar capacitando y diseñar estrategias didácticas potentes que orienten el desarrollo del pensamiento computacional y la enseñanza de la programación en concordancia con los paradigmas vigentes que proponen algo más que la apropiación de una herramienta o de un software. Los procesos educativos con tecnologías implican una forma distinta de significar y de construir una sociedad diferente en la que los dispositivos tecnológicos juegan un papel relevante que es menester problematizar y develar el marco ideológico en que se sostienen. En perspectiva estas

capacitaciones seguirán ofreciéndose acentuando el debate en torno a la soberanía digital, a la creación de herramientas y recursos que propicien las expresiones emancipadoras para movilizar y generar competencias potentes que reclamen entramados sociales más justos e inclusivos.

## Referencias

- Bacher, S. (2007, 12 de agosto) La infancia ¿conectada? Revista La Nación. Disponible en <https://www.lanacion.com.ar/lifestyle/la-infancia-conectada-nid932556/>
- Exeni, C.; González, P. y Melano, A. (2021) Córdoba en pandemia: trabajo docente. Experiencias educativas en tiempos de pandemia. Equipo de docentes - ex becarios Comisión Fulbright -Ed. Ministerio de Educación de la Nación
- Ministerio de Educación de la Nación Argentina (2017) Núcleos de aprendizajes prioritarios de educación digital, programación y robótica. Disponible en: <https://www.educ.ar/recursos/150123/nap-de-educacion-digital-programacion-y-robotica>
- Ministerio de Educación de la Provincia de Córdoba (2015) Diseño curricular del Profesorado de Educación Inicial y Primaria de la Provincia de Córdoba. Disponible en: [https://dges-cba.infed.edu.ar/sitio/curriculares/upload/Disenio\\_Curr\\_Primeria\\_Inicial\\_2015.pdf](https://dges-cba.infed.edu.ar/sitio/curriculares/upload/Disenio_Curr_Primeria_Inicial_2015.pdf)
- Otero, M (2022) Programar para ser ciudadanos libres. Revista Convivimos del 1/05/2022. Disponible en: <https://convivimos.naranja.com/educacion/2022/programar-para-ser-ciudadanos-libres/>
- Umaschi-Bers, M (2016, 6 de mayo) Qué es Scratchjr, la plataforma para enseñarle programación a chicos de 4 a 7 años hecha por una argentina/Entrevistada por Nora Bär para el diario La Nación. Recuperado de [https://www.lanacion.com.ar/tecnologia/chicos-de-4-a-7-anos-aprenden-a-programar-computadoras-con-una-app-desarrollada-por-una-cientifica-argentina-nid1895953/?gclid=Cj0KCQjwhqaVBhCxARIsAHK1tiNOM-HZnKs-WwzRfk8NaAVyCexedNOtmEsA7wdmiPFvkCnyUpfMIK8aAr-YEALw\\_wcB](https://www.lanacion.com.ar/tecnologia/chicos-de-4-a-7-anos-aprenden-a-programar-computadoras-con-una-app-desarrollada-por-una-cientifica-argentina-nid1895953/?gclid=Cj0KCQjwhqaVBhCxARIsAHK1tiNOM-HZnKs-WwzRfk8NaAVyCexedNOtmEsA7wdmiPFvkCnyUpfMIK8aAr-YEALw_wcB)
- Yadav, A. y Berthelsen, U. D. (Eds.). (2021). Computational Thinking in Education: A Pedagogical Perspective. Routledge.



## De la basura electrónica a juegos.

**Autores:** Claudio Javier Di Paolo, [cjdipaolo@gmail.com](mailto:cjdipaolo@gmail.com)

**Resumen.** ¿Qué podemos hacer con la tecnología informática rota u obsoleta de la institución? La mejor respuesta la encontramos en la regla de las 3 R (Reducir, Reciclar y Reutilizar).

Reducir-Reciclar-Reutilizar: Para reducir separamos los componentes de aquellas PC obsoletas o rotas cuyas partes no pueden ni se justifican ser reutilizadas y, con estaciones de soldado/desoldado, desueldan todos los componentes con doble propósito: 1) hacer un stock de componentes electrónicos para reutilizar y 2) todo material desechable se reciclará como juego lúdico o de entretenimiento (damas, ajedrez, ta-te-ti, muñecos, animales, etc); para donar a instituciones que se ocupan de menores de situación vulnerable. Mientras que las computadoras reparadas son equipadas con software libre para su uso.

**Palabras clave:** Reciclar – Reutilizar - Informática – Digitales – Juegos

### Introducción

Marco conceptual en que se basa la experiencia docente

El avance tecnológico nunca se detiene, siempre está en auge, provocando, en voz pasiva o activa conjunciones o disyunciones en la población a partir de sus resultados.

Pero mientras avanza, quedan dispositivos atrás y, por ende, usuarios condicionados por las tecnologías informáticas provocando un impacto social capaz de dividir sectores, de pronunciar desigualdades de oportunidades. Las escuelas no son ajenas a estos cambios, por el contrario, lo padecen. El Instituto Provincial de Educación Media N° 10 Roma, dispone de un parque informático de variada tecnología, como también de espacios de almacenamiento reducidos para albergar dichos equipos. De aquí surge la pregunta ¿Qué podemos hacer para mejorar el orden, clasificar y reducir el espacio ocupado y a su vez aprovechar los recursos?

La respuesta que mejor se ajusta es aplicar la regla de las 3R.

Este concepto hace referencia a estrategias para el manejo de los residuos que se producen todos los días en nuestro hogar o en la industria, buscando ser más sustentables con el ambiente y reducir el volumen de basura generada; lo podemos trasladar a nuestro problema para la búsqueda de soluciones. Entonces, podemos decir conceptualmente que:

Reducir: Nos referimos a modificar el volumen que ocupa un equipo a partir del desarme de sus partes para disponer de ellas, reutilizándolas o bien reciclandolas.

Reutilizar: en consonancia con el ahorro energético y el cuidado ambiental, reutilizar partes de un producto para reemplazar a otro roto, utilizar un producto hasta su fin de vida útil, hacer uso de la imaginación y creatividad en el reuso de componentes para un fin u otro, etc; son ejemplos para aplicar. Reutilizar como insumo o reinventar otro producto.

Reciclar: Cuando estamos frente a un componente o parte o producto totalmente desechable (basura) es donde hablamos de posibilidad de reciclar, es decir, de modificar su razón de existencia para generar otro nuevo producto.

## Descripción de la Experiencia

Contexto, contenidos, metodología, herramientas utilizadas, destinatarios.

### **Contexto:**

El Instituto Provincial de Educación Media N° 10 Roma se encuentra ubicado en la calle Padre Lozano N° 375 del barrio Alto Alberdi, en el Departamento Capital de la provincia de Córdoba.

De la escuela podemos decir:

1. Escuela media con orientación en Informática (1° a 6° año): Aproximadamente 600 estudiantes.
2. Desde 4° año se encuentran cursando los espacios curriculares que conforman la Formación Práctica, uno de ellos es Sistemas Digitales de Información (dispone de 06 hs cátedras semanales distribuidas en dos días). Los 31 estudiantes del 4° “A” y los 28 de 4° “B”, clase a clase trabajan, casi autónomamente, de acuerdo a cada consigna vinculada a los aprendizajes de cada contenido prioritario seleccionado.
3. Con ayuda de máquinas y herramientas (provistas por el Profesor), tales como: una estación de soldado/desoldado, cautín, insumos y otros materiales, los estudiantes se disponen a trabajar en grupo. Al ser colegio medio no técnico se evidencia una carencia de máquinas, herramientas y necesidades que dificultan el acceso a estas prácticas pero que, con voluntad y solidaridad, es posible formarlos en calidad para que logren mejorar su perfil y competencia laboral.

La población está constituida por estudiantes provenientes de familias con realidades y vivencias distintas, con ciertas vulnerabilidades y, otras familias, de clase media. Muchas, carecen de computadoras. El Programa Conectar Igualdad equipó, en gran escala, a las familias del Roma con sus netbooks que, con el tiempo resultan rotas o defectuosas, que también serán parte de nuestra atención.

### **Contenidos:**

A través de un abordaje funcional y sistémico, se propone introducir a los estudiantes en el conocimiento de los aspectos que caracterizan al hardware, o soporte físico (incluye la comprensión de la estructura y organización de las computadoras y de las redes, como así también del modo en que circula, se controla y procesa la información digital). Además, el hardware de los periféricos que interactúan con ellos y que, gracias a los puertos de comunicaciones, la información y su tratamiento fluye de un medio al otro produciendo resultados que satisfacen las necesidades de los usuarios.

Pero todo ello sería en vano si no existiera el software para darle vida y movimiento al hardware. Esta comunión, hardware y software, inseparable y perenne, está presente en cada dispositivo electrónico de información.

Se propone un abordaje histórico que contribuya a desnaturalizar estas tecnologías y, sobre todo, a comprender las lógicas del desarrollo científico-tecnológico, y particularmente, las del desarrollo de informática como disciplina.



La selección de contenidos se orienta a que los estudiantes logren niveles de conceptualización que les permitan tomar decisiones y abordar situaciones problemáticas concretas vinculadas con los sistemas digitales de información, desde una perspectiva que integra hardware y software como así también una amplia gama de tecnologías convergentes. En concreto, que sepan lograr la autonomía a partir de la investigación y desarrollo, para recuperar componentes fundamentales y estructurales ubicados en productos tecnológicos obsoletos y/o en desuso para recuperarlos y usarlos como repuestos (insumos), salvo aquellos que por sus malas condiciones son irrecuperables. Estos últimos son destinados al reciclaje.

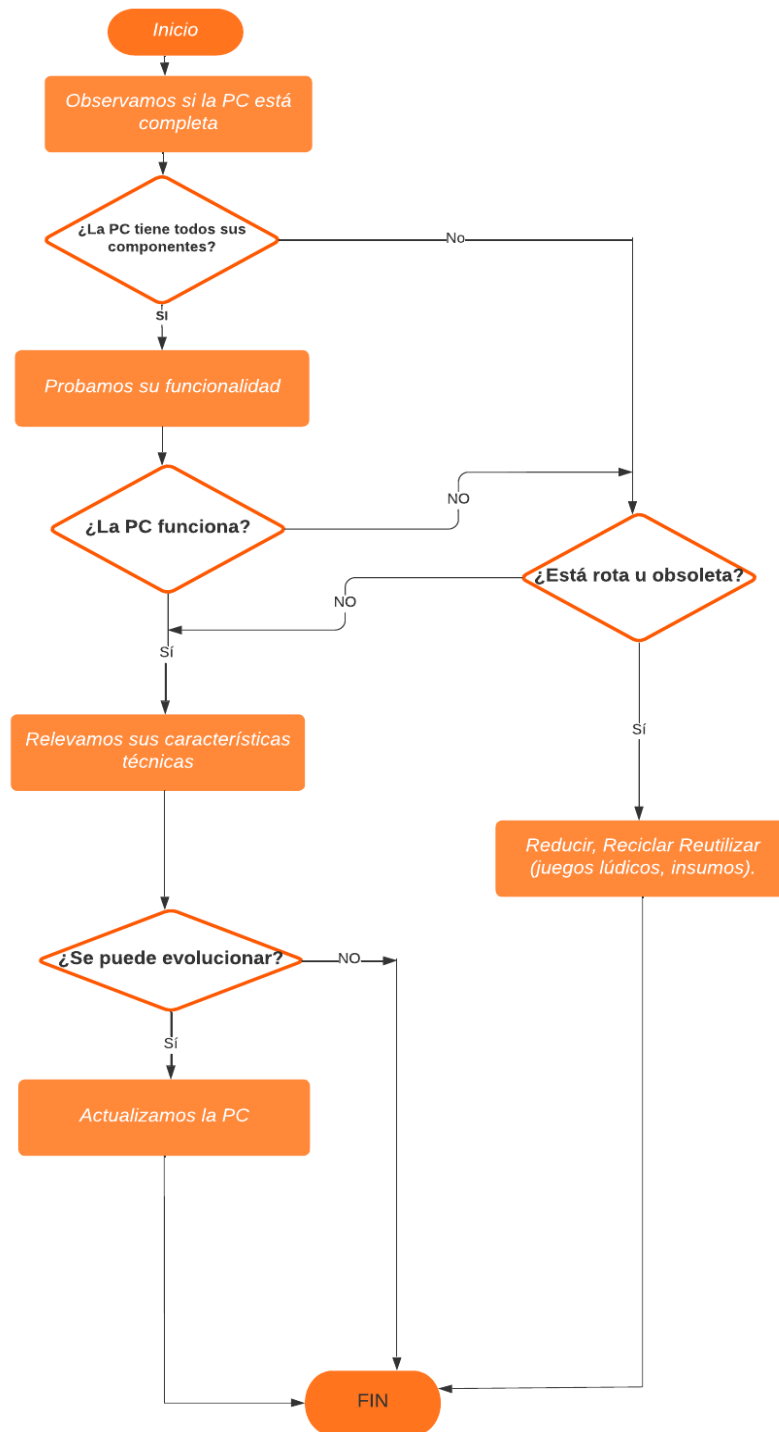
A medida que se avanza en el desarrollo del espacio curricular, se pretende lograr que los estudiantes pueden adquirir los conocimientos necesarios para:

- Investigar y documentar el proyecto.
- Crear un plan de acción para el desarrollo.
- Buscar y seleccionar componentes a partir de la elección del/los juegos a construir.
- Arbitrar los medios para solucionar los problemas.
- Ensamblar las partes.
- Poner en funcionamiento.
- Corregir los problemas y proponer mejoras.

**Metodología:**

Podemos sintetizar la propuesta a partir de un Diagrama de Flujo o de Procesos:

**De la Basura Electrónica a Juegos**



### Descripción del Proceso:






Los estudiantes (en grupos de trabajo) reciben un equipo (comenzamos con PC de escritorios) y continuaremos con equipos portátiles.

Observamos si la PC está completa: En esta primera instancia, los estudiantes ¿serán capaces de observar objetivamente si falta o está mal ensamblado algún componente?

1. ¿La PC tiene todos sus componentes? La decisión tomada a causa de la respuesta deriva en dos caminos distintos:
  - a. Sí, tienen todos sus componentes: ejecuta proceso “Probamos su funcionalidad”
    - i. El proceso “Probamos su funcionalidad” implica conectar a la red eléctrica, los dispositivos fundamentales para probar si enciende y es capaz de cargar el Sistema Operativo.
    - ii. ¿La PC funciona? Con conectarla y encenderla no basta, sino que necesito controlar que cada conexión esté bien hecha.
      1. Sí, funciona ya que enciende y da video, en términos generales este indicador da signos positivos en el equipo dando la posibilidad de ejecutar el proceso “Relevamos sus características técnicas”.
        - a. En “Relevamos sus características técnicas” ocurre gran parte de atención con respecto a los contenidos. Aquí hacemos énfasis en conocer y reconocer a cada parte, su ubicación, su conexión, su función. Sobre todo, incorporamos el concepto de compatibilidad, por ello la pregunta es ¿Se puede evolucionar?
          - i. Si la respuesta es sí, ejecutamos “Actualizamos la PC”:
            1. Actualizamos la PC implica modificar o agregar partes de hardware que aporten una mejora en las variables de optimización del equipo. Pero no sólo evolucionamos un equipo con el hardware sino también cambiando el S.O. por otro superior. En este caso, optamos por probar con distribuciones livianas de Linux para que aprendan su uso básico.
            2. Finalizamos con el equipo en buenas condiciones.
          - ii. Pero si No se puede actualizar, el equipo queda en condiciones limitadas por el hardware y el software que dispone.
        2. Si la PC No funciona, nos preguntamos si ¿Está rota y/o es obsoleta?
          - a. Si la respuesta es NO, recorro nuevamente el camino de “Relevamos sus características técnicas” en búsqueda de una mejora.
    - b. No, les falta componentes. Entonces, pregunto: ¿Está rota y/o es obsoleta? Por cualquiera de estos dos caminos ejecutamos el proceso Reducir, reciclar, reutilizar. Este proceso es eje y fundamento del proyecto porque aquí se manifiesta el que hacer con lo que no sirve. Lejos está de nosotros la opción de “tirar” la basura informática a la basura. Sino que surgen 3 opciones de este proceso:

- *Recuperar* partes o piezas en condiciones reutilizables como insumos para futuras reparaciones: desde dispositivos de almacenamientos, memorias RAM, ROM, cables, gabinetes, microprocesadores, disipadores, coolers, placas madres (funcionales) hasta los componentes electrónicos de placas o dispositivos rotos u obsoletos: transistores, capacitores, bobinas, mosfets, resistencias, etc.
- *Apartar* todo el material desechable, almacenar y luego disponer de él reinventando productos.
- *Reciclar* todas aquellas partes rotas u obsoletas como también componentes de hardware construyendo juegos de entretenimientos, de mesa, de variada creatividad con dos propósitos:
  - *Que adquiera habilidades y destrezas en desoldar componentes como también su reconocimiento y detección de fallas, de tal manera que le ayude en la toma de decisiones.*
  - *Que prevalezca el espíritu solidario poniendo las energías en desarrollar productos de tipo juegos lúdicos para poder donar a quienes más lo necesitan.*
- Finalizamos el Proceso

Los estudiantes trabajando en clases:

		
<p>Desarman y arman PC. Completan y/o recambian componentes.</p>	<p>Desoldando componentes de una motherboard obsoleta. Utilizan estación de soldado por aire caliente.</p>	<p>La PC arrancó. Cambian Sistema Operativo a Distribución compatible de Linux (Bodhi Linux)</p>
<p>Preparan motherboard con Flux, para desoldar componentes.</p>		

Creando juegos lúdicos con las placas desoldadas y con los componentes quitados fuera de circulación.

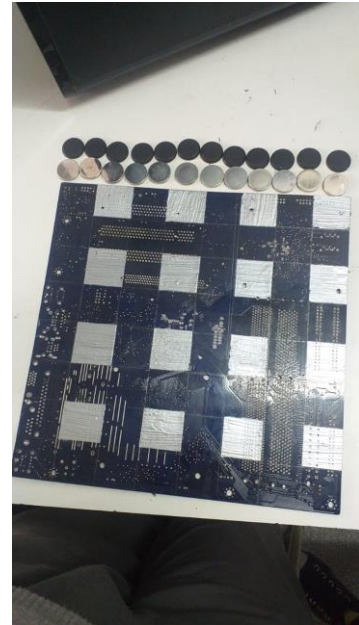
Convirtiendo un viejo teclado en un juego de la Oca



Pintando teclas para hacer un juego de ajedrez.



Juego de Damas (con pilas CR-2032 pintadas y encintadas). Motherboard de tablero.



Y muchos otros juegos lúdicos o de entretenimientos.

También son capaces de hacer pequeños Robots



### ***Herramientas:***

Computadoras con acceso a internet y software necesarios.

Impresora. Hojas. Tinta y/o Tóner

Pizarrón. Marcadores. Proyector. Parlantes.

Tutoriales. Material de referencia para la lectura y el ejemplo.

Impresoras, escáneres, computadoras, tablets, etc, en desuso y/u obsoletas para recuperar componentes.

Espacio destinado a taller para el desarme y armado de equipos con bancos de trabajo.

Estación de soldado (aire caliente y cautín) - Fuente reguladora de energía (2A o 3A) - Flux - Pasta térmica - Estañolín - Destornilladores - Pinzas y alicates (chicos) - Aerosol “Detector de fallas” - Aerosol Aire comprimido - Alcohol isopropílico - Multímetro - Soporte para soldado -

## **Resultados**

Aprendizajes logrados. Fortalezas y debilidades identificadas.

La experiencia es positiva, rompe estereotipos de aulas en donde el Profe expone y los estudiantes prestan atención. Aquí el Profe está al servicio del estudiante quien demanda su atención. Es cierto que se produce “un poco de ruido” en la comunicación dentro del aula, por las conversaciones entre los estudiantes como también el ruido propio de algunas herramientas trabajando. Crear en el estudiante una conciencia ecológica, sobre todo en lo que a Basuras Electrónicas se refiere, comprender que tirar está prohibido y que reciclar, con el fin reutilizarlos, a partir de una reinención de productos, es una alternativa que no solo aporta mejoras a la ecología sino también a la sociedad.

### ***Aprendizajes logrados***

El proyecto está en ejecución, en un 60%, y se espera que:

- Adquieran destrezas y habilidades en el proceso de armado y desarmado de PC (la mayoría manifestó que nunca había abierto ni desarmado una PC)
- Desarrollar capacidades para realizar tareas de instalación, conectando y configurando componentes.
- Identificar y caracterizar periféricos y componentes, analizar incompatibilidades y posibles causas de fallas, como también encontrar la solución adecuada al mejor costo.
- Interpretar información técnica.
- Desarrollar la capacidad de prever posibles problemas durante la instalación de software.
- Apliquen creatividad para reinventar productos, por ejemplo, juegos, a partir de la basura tecnológica producto de la obsolescencia de una computadora.
- Aprendan a desoldar y a soldar componentes electrónicos en una placa.
- Aprendan a testear una fuente de alimentación y a medir sus valores con el multímetro.

### ***Fortalezas:***

Las fortalezas observadas en el desarrollo de cada práctica son:

- La motivación y el interés por hacer.
- La ansiedad por aprender y a su vez la paciencia por esperar una respuesta (atenciones personalizadas).
- La curiosidad de descubrir que hay adentro del gabinete.
- La necesidad de aprender implica compromiso y estudio.
- El respaldo de la Gestión Institucional en la dinámica implementada en el aula.
- El reconocimiento de los componentes a partir de la exploración de una placa madre y su extracción.

### **Debilidades:**

Las debilidades manifestadas, tanto en los estudiantes como en la institución, para llevar adelante el proyecto son:

- Crear el hábito óptimo de trabajo en los estudiantes. (Al ser una Escuela Media la cultura del trabajo en un Taller-Laboratorio no está instalada).
- Carecer de un espacio de trabajo acorde a la práctica. (Simulamos al aula como un espacio productivo).
- Comprometer equitativamente a cada estudiante para que cumplan roles rotativos en forma autónoma.
- Fortalecer la capacidad de independencia y autosuficiencia en la resolución de problemas.
- Incorporar más lecto-comprensión de textos que le brinden aportes teóricos.

### **Conclusiones**

Propuestas superadoras basadas en la experiencia para continuar a futuro.

Implementar actividades meramente prácticas, en donde el Hacer prevalece generando necesidad del Saber, manifiesta al aprendizaje y a la cultura del Ser de un estudiante frente a un desafío técnico. Si bien la escuela no forma técnicos sino a estudiantes con Orientación en Informática, es necesario y complementario que conozcan a una computadora, que trabajen en equipo, que sean capaces de detectar fallas y proponer soluciones para lograr ser más competentes en el mundo laboral o simplemente para uso personal.

Que el estudiante empiece a generar el compromiso de ejecutar una tarea asignada en forma autónoma, provoca cambios culturales y sociales. Muchos de los estudiantes desconocían como utilizar un destornillador ni identificaban a un alicate (lo confundían con una pinza) porque no se les da oportunidad de aprender desde lo práctico y, peor aún, no les enseñan lo básico en la casa. Descubrir los tipos de corriente y sus voltajes, hacer mediciones, testear una fuente de alimentación, aprender a enchufar los conectores en cada puerto, son vivencias que provocan alegría cuando confirman que lo que hicieron sirvió, salió bien y arrancó “la compu”. Y si el estudiante se pone feliz por aprender, “el profe” inmensamente feliz.

Esta dinámica requiere una evaluación diaria, en donde se valore el desempeño recurriendo a la observación y el diálogo, como medio de relevamiento, para medir su actitud, aptitud, iniciativa, interés, autonomía, rendimiento y proactividad.

### **Referencias**

[Tablas de comparación para la familia de procesadores... - Intel](https://www.intel.la > www > support > articles > processors)  
<https://www.intel.la > www > support > articles > processors>

Diferencia entre i9, i7, i5 e i3 en **procesadores Intel® Core™** de 10<sup>a</sup>, 9<sup>a</sup>, 8<sup>a</sup> y 7<sup>a</sup> Generación para desktops

### [Procesadores de escritorio AMD](#)

<https://www.amd.com> › products › processors-desktop

**Procesadores** para computadoras de escritorio empresariales y comerciales · **Procesadores AMD Ryzen™ Threadripper™ PRO** · **Procesadores AMD Ryzen™ PRO** · **Procesadores**

### [Cómo encontrar placas base compatibles con su procesador...](#)

<https://www.intel.la> › processors › intel-core-processors

La Herramienta de **compatibilidad** de Intel® identifica la **compatibilidad** entre los procesadores para computadoras de escritorio de Intel® y **placas** base para ...

### [Especificaciones de placas madre | Centro para socios de AMD](#)

<https://www.amd.com> › partner › motherboards

Todos los **procesadores** de **AMD** necesitan una **placa madre**. Utiliza nuestra herramienta a continuación para ver **placas madre** en una lista comparativa, ...



## Identificación de sensores utilizados en robótica educativa mediante aprendizaje automático

Javier Berger<sup>1 2</sup>, Solange Schelske<sup>1 2</sup>

<sup>1</sup>Escuela de Robótica Misiones  
Av. Roca 480, Posadas (3300), Misiones, Argentina

<sup>2</sup> Doctorando en Informática, Facultad de Ciencias Exactas, Químicas y Naturales. Universidad Nacional de Misiones. Félix de Azara 1552, Posadas (3300), Misiones, Argentina  
{javiberger, ingsolangeschelske}@gmail.com

**Resumen.** Los sensores que integran los kits de robótica educativa representan un componente fundamental en la enseñanza de robótica, mediante estos se miden distintas magnitudes del entorno. Cuando se trabaja con estos componentes, se requiere identificar qué sensor del kit es el adecuado para la medición a realizar. La tarea de identificación es compleja ya que las características visuales de los sensores se asemejan y esto provoca que, quién no tenga el entrenamiento visual suficiente para diferenciarlos, se confunda en la selección del sensor. En este trabajo se propone la aplicación de técnicas de aprendizaje automático y procesamiento de imágenes con el objetivo de generar una herramienta para la clasificación de siete sensores incluidos en los kits de robótica educativa. Para esto, se generó un conjunto de datos propios con imágenes de siete sensores y se entrenó el modelo de red neuronal convolucional VGG16 mediante aprendizaje por transferencia. Con el modelo entrenado se obtuvo 97,14 % de aciertos, para la métrica F1, en la clasificación de imágenes de sensores de un conjunto de test. Finalmente, se desarrolló una API que recibe la imagen de un sensor y retorna una etiqueta con la identificación obtenida para la misma mediante el modelo entrenado.

**Palabras clave:** aprendizaje automático, procesamiento de imágenes, redes neuronales convolucionales, robótica educativa, sensores.

### 1 Introducción

El sistema educativo se enfrenta, permanentemente, a distintos desafíos entre los que se encuentra el de adaptar los procesos de enseñanza y aprendizaje a las nuevas tecnologías que irrumpen en el aula y en la sociedad en general (Massei et al., 2019). En Argentina, durante la última década, se instaló el concepto de robótica dentro de las instituciones educativas dando lugar a la generación de innumerables proyectos vinculados con la temática (Queiruga et al., 2021).

La robótica pertenece al área de las ciencias de la computación e involucra la aplicación de conocimientos relacionados con programación, electrónica, tecnología de sensores, inteligencia artificial, entre otros. En este sentido y como se define en (Banchoff Tzancoff et al., 2019), “*la robótica educativa se convierte en un recurso para facilitar el aprendizaje y desarrollar competencias generales como la socialización, la creatividad y la iniciativa, que permitan al estudiante dar respuesta eficiente a los entornos cambiantes del mundo actual*”.

Dentro de la robótica educativa el uso de sensores juega un papel fundamental en el aprendizaje y en el desarrollo de proyectos. Mediante estos componentes, los estudiantes comprenden cómo se percibe el mundo físico dentro del campo de la robótica (Ramírez et al., 2014). Es así que, permanentemente, surgen nuevos sensores que permiten aplicar robótica a nuevas áreas y, en paralelo, se mejoran las capacidades de los sensores existentes. Además, el impulso de la robótica educativa, conlleva a la disponibilidad de sensores de bajo costo que permiten a los estudiantes disponer de los mismos dentro de las instituciones educativas e implementarlos en distintas soluciones vinculadas con robótica.

En las instituciones educativas que implementa robótica educativa se observa, la presencia de los denominados kits de robótica. Un kit se compone de distintos tipos de sensores, como así también de actuadores relacionados, mayoritariamente, con la plataforma de hardware y software libre Arduino (Candelas-Herías et al., 2015). La mencionada plataforma es un actor fundamental en la implementación masiva de robótica educativa, ya que los elementos que integran los distintos kits basados en Arduino son de uso sencillo y de bajo costo.

La masificación del uso de sensores para la plataforma Arduino, como así también la filosofía del hardware libre, potencia el surgimiento de innumerables fabricantes que desarrollan sensores para la medición de una gran variedad de magnitudes físicas. Los sensores de los distintos fabricantes poseen características únicas y formas diferentes, con el objetivo de facilitar la aplicación de los mismos en proyectos de robótica educativa.

En instituciones educativas dedicadas exclusivamente a la enseñanza de robótica, como es el caso de la Escuela de Robótica Misiones (Silvero & Escalada, 2019), se observa la disponibilidad de sensores de distintos fabricantes, ya que dentro de la mencionada institución se valida el funcionamiento de estos sensores para, posteriormente, replicar el uso en otras instituciones. La existencia de una gran variedad de sensores acarrea una problemática que se relaciona con identificación de los mismos por parte de estudiantes, facilitadores y personal administrativo encargado del resguardo de los diferentes componentes. Esto se fundamenta en que existen sensores que a primera vista son muy similares, a esto se le suma que dependiendo del fabricante se utilizan distintas siglas identificatorias o incluso nombres específicos que se confunden entre fabricantes. Es así que, por los motivos mencionados, en ocasiones existen sensores que se encuentran disponibles en los espacios de trabajo pero que los estudiantes no los utilizan por desconocimiento de las especificaciones del mismo. Asimismo, otra situación se da cuando los facilitadores elaboran el pedido de componentes electrónicos a los responsables administrativos y dentro del mismo se detallan sensores específicos, en ocasiones, el facilitador a la hora de utilizar lo solicitado en el aula se encuentra con sensores que no correspondían con los solicitados previamente.

La identificación de objetos y clasificación de los mismos mediante el uso de imágenes es un área de estudio en auge gracias a los avances alcanzados en el campo del aprendizaje automático. Específicamente mediante el uso de redes neuronales convolucionales disponibles para ser entrenadas en diferentes plataformas gratuitas que facilitan la creación de sistemas de identificación (Rusiñol & Gómez, 2018). De esta manera, se logran soluciones de identificación de elementos a partir del uso de imágenes en diferentes campos como ser la agricultura (Leonardo & Faria, 2019) (Seijas et al., 2019), la industria (Yaranga et al., 2018) (Ortiz de Zúñiga Mingot, 2020) y salud (Wolyniec Rojas, 2020) (Orosco, 2019).

Por los motivos anteriormente mencionados, se plantea el uso de técnicas de aprendizaje automático y procesamiento de imágenes para el desarrollo de una solución que permita mediante una imagen identificar qué sensor se está observando. De esta manera, los estudiantes pueden hacer uso de la solución para identificar los sensores disponibles en los espacios de trabajo y, también, los administrativos pueden confirmar que los componentes electrónicos preparados se corresponden con la solicitud recibida.

En la siguiente sección se detallan las características principales del aprendizaje automático que fundamentan este trabajo. En la sección 3, se detallan los materiales y métodos utilizados para el desarrollo de la solución propuesta. En la sección 4 se analizan los resultados obtenidos y en la sección 5 se presentan las conclusiones y trabajos futuros.

## 2 Aprendizaje automático

Las mejoras recientes vinculadas con la clasificación de imágenes y el reconocimiento de objetos se relacionan directamente con los avances logrados dentro del campo del deep learning o aprendizaje profundo (Lecun et al., 2015). Este subcampo del aprendizaje automático y de la inteligencia artificial tiene como objetivo generar sistemas capaces de emular las capacidades que, hasta no hace muchos años, se les atribuía únicamente a los humanos, tal es el caso de la capacidad de identificar un sensor a partir de una única imagen.

Detrás del avance del aprendizaje profundo para el procesamiento de imágenes y video se encuentran las redes neuronales convolucionales (CNN). Este tipo de redes neuronales artificial intentan imitar el comportamiento humano con el objetivo de reconocer las características más importantes que permiten describir un objeto (Krizhevsky et al., 2012). Utiliza una jerarquía de capas ocultas especializadas en distintas actividades de reconocimiento. Así, las capas exteriores se encargan de identificar, por ejemplo, las líneas y curvas dentro de una imagen, mientras las capas más profundas se dedican a reconocer el contorno de un sensor.

El uso de CNN para la clasificación de imágenes se relaciona, habitualmente, con la necesidad de contar con miles de imágenes de ejemplo para entrenar un modelo y, posteriormente, lograr que el mismo clasifique correctamente nuevas imágenes. Esto provoca que la obtención de un conjunto de datos de imágenes para entrenar los modelos de clasificación se vuelva una tarea compleja ya que el acceso a estos datos se encuentra restringido o existen muy pocos ejemplos para utilizar en el modelado de la CNN.

Para encarar la problemática anteriormente mencionada surge el aprendizaje por transferencia (Weiss et al., 2016), esta técnica permite utilizar un modelo de CNN previamente entrenado para la solución de un problema de clasificación de imágenes y aplicarlo en el desarrollo de un clasificador para otro problema de clasificación con características similares. De esta manera, se utiliza el aprendizaje de un modelo que ya tiene la capacidad de clasificar correctamente imágenes y se realiza un reentrenamiento para ajustar las últimas capas. Con esto se logra acelerar el proceso de desarrollo de modelos de clasificación, se reduce la necesidad de contar con grandes cantidades de imágenes y brinda la posibilidad de utilizar equipos que no poseen gran capacidad de cómputo, ya que la tarea más costosa en cuanto a recursos se realiza durante el entrenamiento del modelo inicial (Lu et al., 2015).

En paralelo a la generación de nuevos modelos que facilitan el desarrollo de clasificadores también se expande día a día el uso de librerías que facilitan el desarrollo de modelos de clasificación. Entre estas se destaca Tensorflow, la biblioteca de código abierto de Google que utiliza gráficos de flujo de datos para representar cálculos (Abadi et al., 2016). Con el lanzamiento de Tensorflow versión 2.0 se integró en la misma Keras, un framework de alto nivel para el desarrollo modelos de aprendizaje profundo escrito en Python ampliamente utilizado por investigadores del área (Ketkar & Ketkar, 2017).

Las diferentes librerías que implementan aprendizaje por transferencia contienen distintas redes conocidas por su capacidad de predicción de grandes conjuntos de imágenes como es el caso del conjunto *imagenet* compuesto por 14 millones de imágenes de 20.000 categorías distintas (Deng et al., 2009). Entre los modelos preentrenados contenidos en Keras se encuentra: VGG16, VGG19,

ResNet50, InceptionV3, InceptionResNetV2, MobileNet, MobileNetV2, DenseNet y NASNet. De estas redes se destaca VGG16, propuesta por Karen Simonyan y Andrew Zisserman en su artículo *Very Deep Convolutional Networks for Large-Scale Image Recognition* (Simonyan & Zisserman, 2014), por contar con una estructura secuencial sencilla de 16 capas y obtener un 92,7 % de clasificaciones correctas para el conjunto de datos de *imagenet*.

Como se aprecia en la Fig. 1, en VGG16 la entrada a la capa convolucional uno es la entrada de una imagen RGB de tamaño fijo de 224 x 224. Esta imagen pasa a través de una pila de capas convolucionales donde los filtros se usan con un campo receptivo de 3 x 3. Asimismo, se utilizan filtros de convolución 1 x 1, que se pueden ver como una transformación lineal de los canales de entrada. Por otra parte, la agrupación espacial se lleva a cabo mediante cinco capas de agrupación máxima, que siguen a algunas de las capas convolucionales. Posee tres capas totalmente conectadas seguidas de una pila de capas convolucionales hasta llegar a la última capa que es de tipo softmax. Todas las capas ocultas utilizan la no linealidad de rectificación, conocida como Relu (Bhagwat et al., 2019).

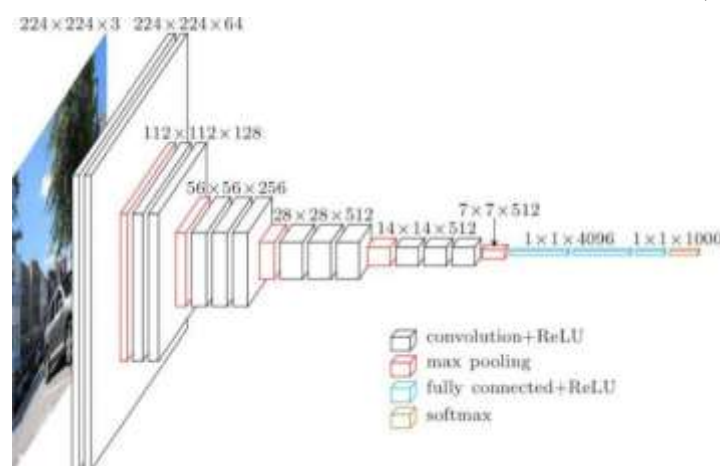


Fig. 1: Arquitectura VGG16. Fuente: *neurohive.io*

### 3 Materiales y métodos

Los sensores utilizados para este trabajo se obtuvieron del área administrativa de la Escuela de Robótica Misiones. Se seleccionaron, como se aprecia en la Tabla 1, siete sensores, que, según los responsables de la distribución de componentes son requeridos frecuentemente y por sus características se confunden con otros sensores de formas similares.

Tabla 1: Sensores utilizados

Identificador del fabricante	Nombre utilizado
KY002	Sensor de Shock
KY006	Zumbador Pasivo
KY010	Sensor Fotoeléctrico Tipo U
KY011	Modulo LED Bicolor 5MM
KY021	Interruptor Magnético Mini
KY035	Sensor de campo magnético
KY039	Sensor Pulso Cardíaco

Con los sensores seleccionados se generó un conjunto de imágenes, para esto se utilizó un dispositivo móvil LG G4 que cuenta con una cámara de 16 megapíxeles. El dispositivo se montó sobre un soporte que cuenta con iluminación artificial mediante una lámpara LED de 9W, como se observa en

la Fig. 2. Con esta configuración, todas las capturas obtenidas guardaban la misma relación de distancia. El sensor se colocó sobre dos tipos de fondo, uno correspondiente a un fondo blanco y otro a un fondo marrón, con el objetivo de representar distintos entornos en los que se puede consultar sobre el sensor.

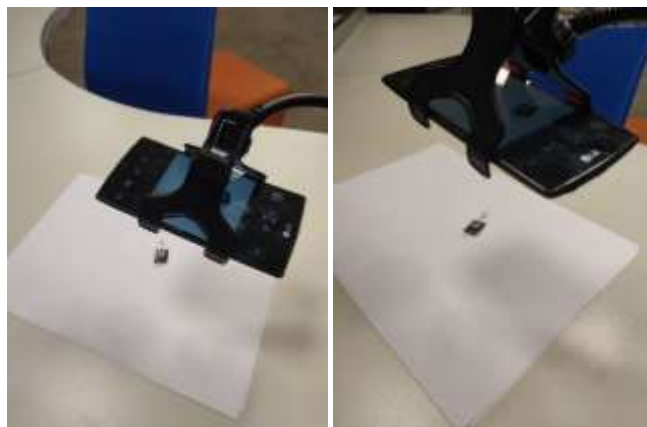


Fig. 2: Captura de imágenes de sensores.

Las imágenes obtenidas se redimensionaron a 224 x 224 píxeles a partir de los originales 2976 x 2976 para facilitar el procesamiento de las mismas mediante las técnicas de aprendizaje automático.

Se obtuvo un conjunto de 280 imágenes, constituido por 40 imágenes de cada uno de los siete sensores detallados en la Tabla 1. En la Fig. 3 se aprecian imágenes de muestra de los sensores utilizados.



Fig. 3: Imágenes de muestra de los sensores

De las 280 imágenes disponibles, se utilizaron, como se aprecia en la *Tabla 2*, 210 para el entrenamiento del modelo, 35 para validación y 35 para test.

Tabla 2: Distribución de las imágenes

Sensor	Entrenamiento	Validación	Test
KY002	30	5	5
KY006	30	5	5
KY010	30	5	5
KY011	30	5	5
KY021	30	5	5
KY035	30	5	5
KY039	30	5	5
<b>TOTAL</b>	<b>210</b>	<b>35</b>	<b>35</b>

Para la generación del modelo se utilizó el sistema de gestión de paquetes Anaconda instalado en una notebook con Windows 10 de 64bits que cuenta con procesador Intel i5 – 10210U de 2.11 GHz con gráfica integrada, 8 GB de memoria RAM DDR4 y disco SSD NVMe de 240 GB. Dentro de Anaconda se utilizó Jupyter Notebook con el lenguaje de Programación Python 3.7, las librerías Tensorflow 2.1, Keras 2.3 y scikit-learn.

Durante el entrenamiento del modelo se utilizó *ImageDataGenerator* disponible en Keras, mediante el mismo se aumentó de forma aleatoria el conjunto de imágenes utilizando rotación de 20 grados, zoom de 0.2, cambio de ancho y altura de 0.1 y, rotaciones horizontales y verticales.

El entrenamiento se realizó aplicando aprendizaje por transferencia con base en la red neuronal convolucional VGG16 con los pesos preentrenados para el conjunto de datos *imagenet*. Al mismo se le modificó la última capa para adecuar a la estructura del clasificador de sensores, para esto se le agregó una capa densa que tiene como salida el número de sensores a clasificar con una activación *softmax* y una métrica de entrenamiento basada en la precisión.

Concluida la etapa de entrenamiento se guardó el modelo entrenado para utilizarlo en una API desarrollada con el framework de desarrollo web Flask para correr el modelo en el entorno local. Para esto se generó una página de consulta en la que el usuario puede cargar una imagen y mediante la API obtener la predicción correspondiente.

## 4 Resultados

El entrenamiento del modelo VGG16, pre entrenado con el conjunto de imágenes de *imagenet*, utilizando aprendizaje por transferencia con las imágenes de los siete sensores de la Tabla 1, tuvo una duración de 328 segundos, un promedio, como se observa en la Fig. 4, de 33 segundos por cada una de las 10 épocas con la que se configuró.

```
Epoch 1/10
8/8 [=====] - 34s 4s/step - loss: 2.4009 - accuracy: 0.2432 - val_loss: 1.2058 - val_accuracy: 0.4800
Epoch 2/10
8/8 [=====] - 30s 4s/step - loss: 1.2816 - accuracy: 0.5351 - val_loss: 0.5702 - val_accuracy: 0.9000
Epoch 3/10
8/8 [=====] - 33s 4s/step - loss: 0.5976 - accuracy: 0.8486 - val_loss: 0.6534 - val_accuracy: 0.8000
Epoch 4/10
8/8 [=====] - 30s 4s/step - loss: 0.6823 - accuracy: 0.8162 - val_loss: 0.1589 - val_accuracy: 1.0000
Epoch 5/10
8/8 [=====] - 35s 4s/step - loss: 0.3789 - accuracy: 0.8950 - val_loss: 0.3310 - val_accuracy: 0.9600
Epoch 6/10
8/8 [=====] - 35s 4s/step - loss: 0.3154 - accuracy: 0.9189 - val_loss: 0.2745 - val_accuracy: 0.9000
Epoch 7/10
8/8 [=====] - 37s 5s/step - loss: 0.3230 - accuracy: 0.9297 - val_loss: 0.3643 - val_accuracy: 0.8400
Epoch 8/10
8/8 [=====] - 31s 4s/step - loss: 0.3292 - accuracy: 0.9135 - val_loss: 0.1749 - val_accuracy: 1.0000
Epoch 9/10
8/8 [=====] - 33s 4s/step - loss: 0.2331 - accuracy: 0.9622 - val_loss: 0.2221 - val_accuracy: 0.9600
Epoch 10/10
8/8 [=====] - 30s 4s/step - loss: 0.1859 - accuracy: 0.9622 - val_loss: 0.1351 - val_accuracy: 1.0000
```

Fig. 4: Entrenamiento del modelo

Mediante *plotTraining* se observó el historial de entrenamiento en el que se registran las métricas para cada una de las épocas. Esto incluyó, como se observa en la Fig. 5, la precisión y la pérdida para los conjuntos de entrenamiento y validación. Mediante el mismo se observó que el modelo supero el 96% de precisión para el conjunto de entrenamiento en la época nueve, en tanto que para el conjunto de validación este valor se alcanzó en la época cinco. En tanto que la pérdida llegó al mínimo tanto en el conjunto de entrenamiento como de validación en la época diez.

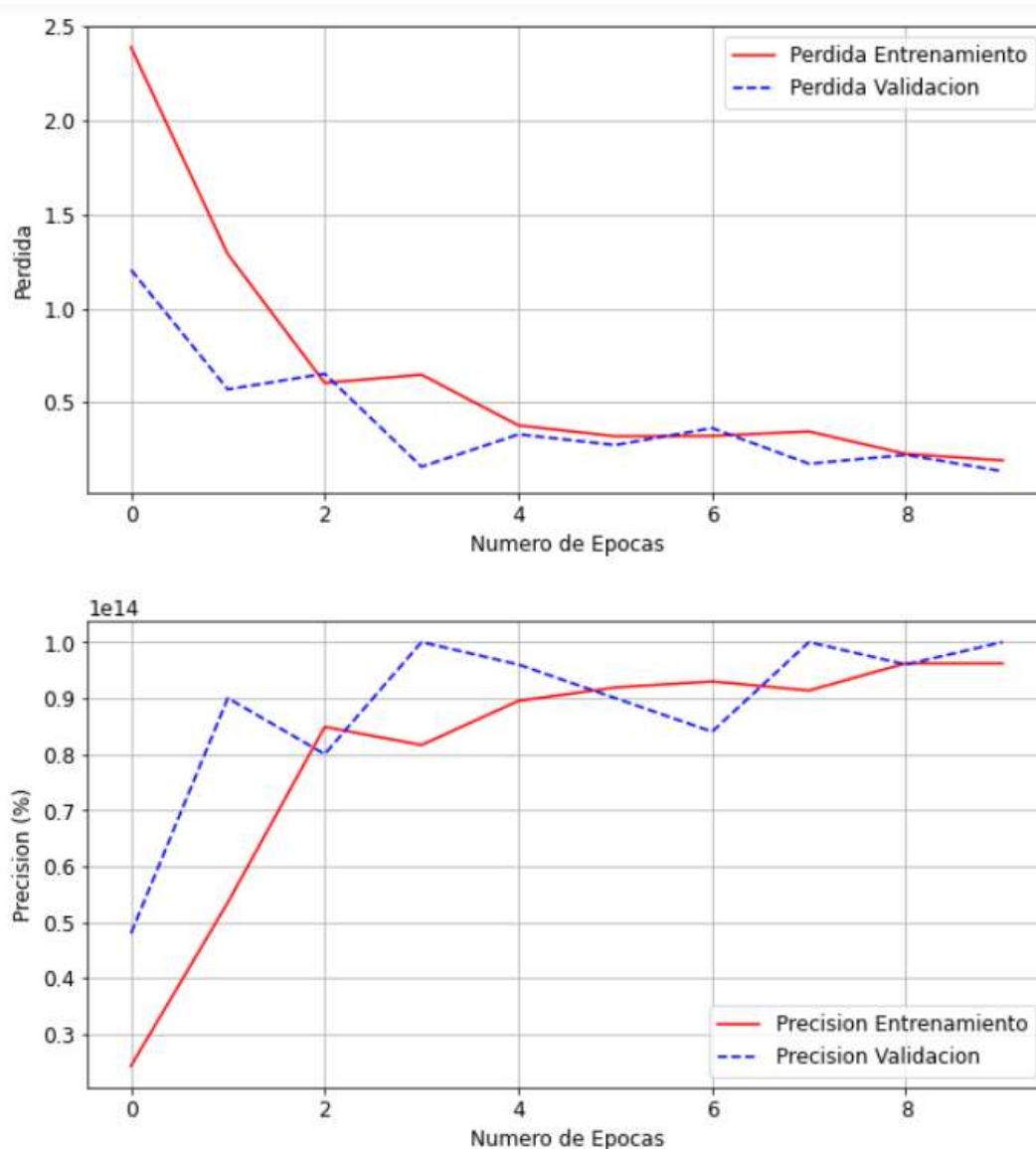


Fig. 5: Gráficas del historial de entrenamiento

Con el modelo entrenado se evaluaron las imágenes del conjunto de test, para esto se utilizó la matriz de confusión con el objetivo de obtener las métricas para las distintas clases de objetos clasificados. Como se observa en la Fig. 6, un sensor KY002 fue etiquetado como KY021 por el modelo, las demás predicciones fueron correctas. En función de lo anterior y como se observa en la Fig. 7, para los sensores KY002, KY006, KY010, KY011, KY035, KY039 se obtuvo el 100% de precisión, en tanto que para el sensor KY021 se obtuvo 83,3% de precisión. En tanto que en relación a la exhaustividad para los sensores KY006, KY010, KY011, KY021, KY035, KY039 se obtuvo un 100 % en tanto que para el KY002 un 80%. En función de los valores mencionados se obtuvo para la métrica F1 un 100% en los sensores KY006, KY010, KY011, KY035, KY039, un 88,89 % en el sensor KY002 y un 90,91 % KY021 completando un total de 97,14 % para el total de muestras de entrenamiento.



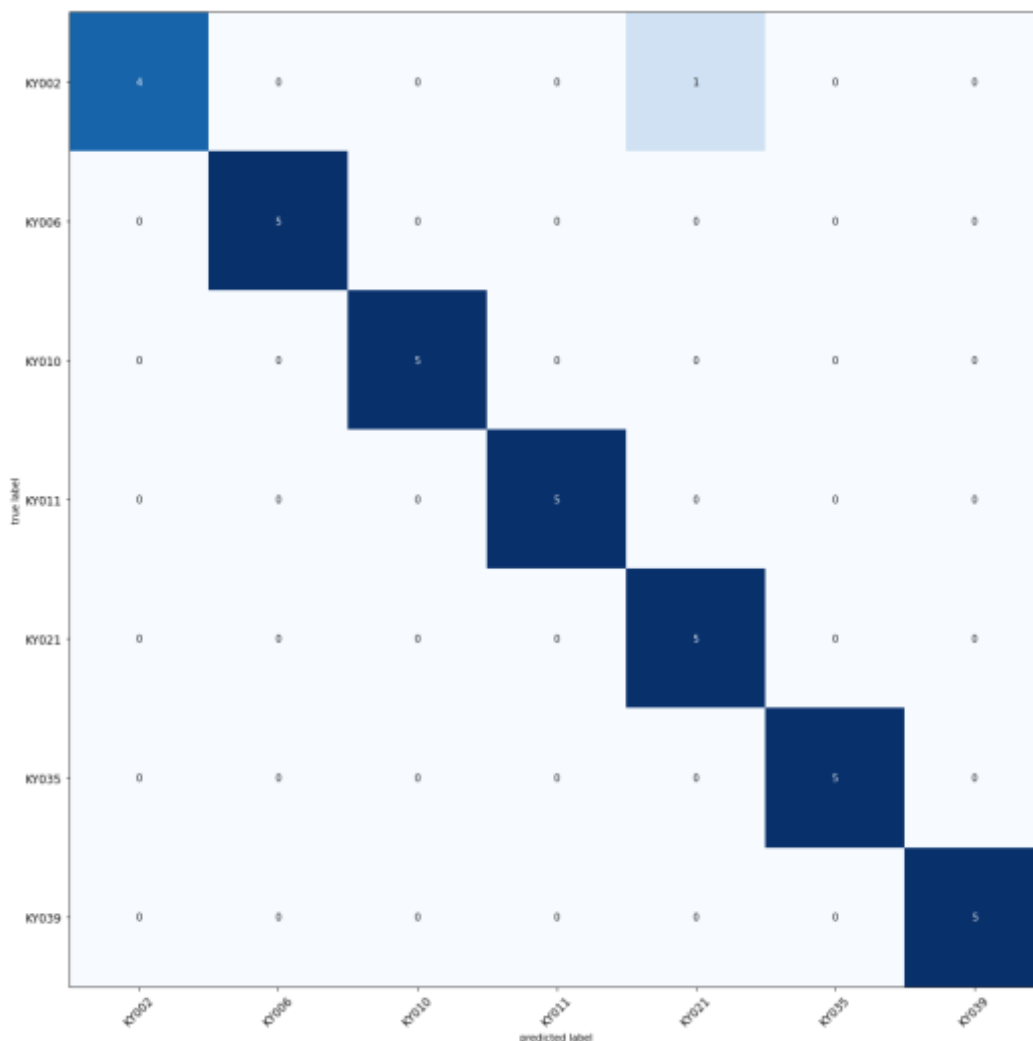


Fig. 6: Matriz de confusión

```

Found 35 images belonging to 7 classes.
      precision    recall  f1-score   support

   0:   1.0000    0.8000    0.8889     5
   1:   1.0000    1.0000    1.0000     5
   2:   1.0000    1.0000    1.0000     5
   3:   1.0000    1.0000    1.0000     5
   4:   0.8333    1.0000    0.9091     5
   5:   1.0000    1.0000    1.0000     5
   6:   1.0000    1.0000    1.0000     5

 accuracy          0.9714     35
 macro avg         0.9762     35
 weighted avg     0.9762     35
    
```

Fig. 7: Métricas

Con los resultados obtenidos para el conjunto de test se continuó con la siguiente etapa que consistió en el desarrollo de la API utilizando Flask. Así, se desarrolló el aplicativo *identificar\_sensor.py* en el que se cargaron las librerías necesarias para utilizar el modelo a partir de una imagen subida a



desde un navegador web y mediante un botón se carga la imagen y pasa la misma a la API que se encuentra ejecutando en el entorno local, finalmente esta retorna el valor de predicción para la imagen utilizada. En la *Fig. 8* se observa la pantalla inicial en la que se solicita la carga de una imagen de un sensor, en tanto que en la *Fig. 9* se observa la imagen cargada a predecir y botón para solicitar la identificación, finalmente en la *Fig. 10* se observa la predicción y, además cómo responde la API a la solicitud de predicción.



Fig. 8: Pantalla inicial



Fig. 9: Imagen cargada

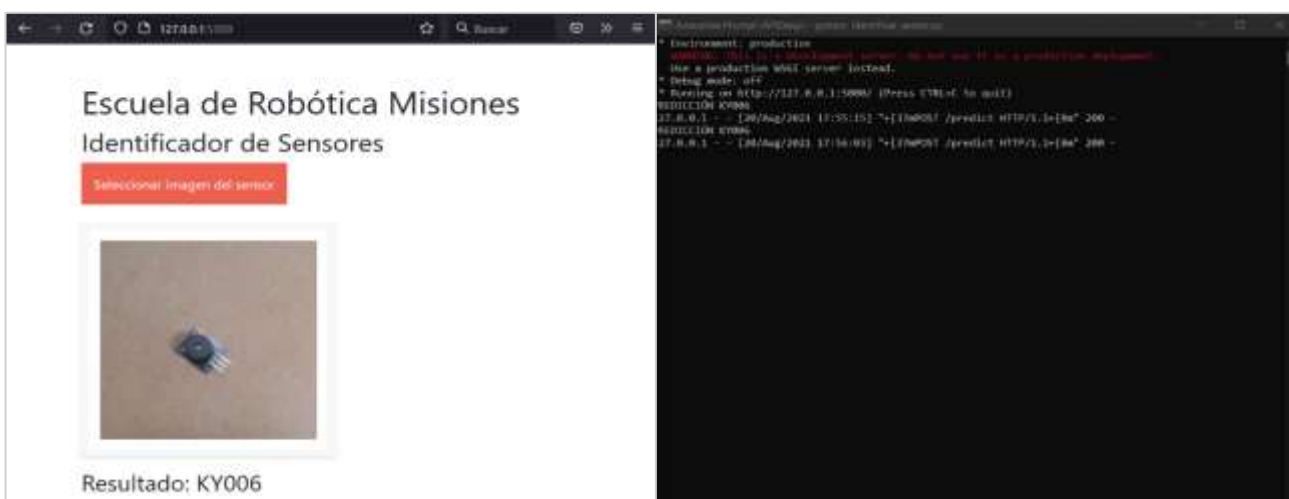


Fig. 10: Predicción obtenida

## 5 Conclusiones

Este trabajo se desarrolló en el contexto de la aplicación de técnicas innovadoras para el desarrollo de soluciones aplicadas a la enseñanza de robótica educativa en la Escuela de Robótica Misiones.

En este trabajo se generó un conjunto de imágenes para siete sensores utilizados en robótica educativa, que se detallan en la Tabla 1. En función del conjunto de datos obtenido y utilizando técnicas de aumento de datos se entrenó un modelo de red neuronal convolucional basado en VGG16 aplicando aprendizaje por transferencia en función de los pesos del conjunto de datos *imagenet*.

El proceso de entrenamiento se completó en 328 segundos, utilizando una notebook con características estándar, es decir que no cuenta con hardware específico para el procesamiento de grandes volúmenes de imágenes, como es el caso de tarjetas gráficas dedicadas. Como resultado del entrenamiento se obtuvo un modelo con la capacidad de identificar, en función de la métrica F1, un 97,14 % de las muestras del conjunto de test. Si bien es posible entrenar el modelo en entornos en la nube, como el caso de Google Colab, mediante este trabajo se demostró que utilizando aprendizaje por transferencia es posible generar modelos con suficiente capacidad de clasificación en equipos estándar.

Con los resultados obtenidos se desarrolló una API en un entorno local que permite subir una imagen de un sensor del grupo de estudio y obtener la predicción correspondiente para la misma.

Esta solución constituye una herramienta innovadora en la tarea de identificación de componentes de kit para robótica educativa, facilitado la tarea de reconocimiento de los distintos sensores que se incluyen en los kits.

Si bien el modelo logró un nivel superior al 90 % en las clasificaciones, se observó que para mejorar la clasificación del mismo sería necesario el uso de un conjunto de datos mayor, como así también realizar pruebas con otros modelos de redes neuronales convolucionales previamente entrenadas.

Por otra parte, como trabajo futuro se plantea la necesidad de incorporar más sensores a clasificar, como así también implementar la API en un servidor disponible en internet para que estudiantes, facilitadores y administrativos de la Escuela de Robótica Misiones, como de otras instituciones puedan hacer uso del clasificador entrenado. De esta manera, el impacto que generaría la solución presentada, en las experiencias educativas de las instituciones que hacen uso de kits de sensores para la enseñanza de robótica, permitiría agilizar el desarrollo de proyectos de robótica mediante la identificación rápida e inequívoca de cada sensor a utilizar.

## 6 Referencias

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., ... Zheng, X. (2016). *TensorFlow: A System for Large-Scale Machine Learning*.
- Banchoff Tzancoff, C. M., Martín, E. S., Gómez, N. S., & López, F. E. M. (2019). Experiencias en robótica educativa: diez años trabajando con escuelas argentinas. *XIV Congreso Nacional de Tecnología en Educación y Educación en Tecnología (TE&ET 2019)(Universidad Nacional de San Luis, 1 y 2 de julio de 2019)*.
- Bhagwat, R., Abdolahnejad, M., & Moocarme, M. (2019). *Applied Deep Learning with Keras: Solve complex real-life problems with the simplicity of Keras*. Packt Publishing. <https://books.google.com.ar/books?id=6--UDwAAQBAJ>
- Candelas-Herías, F. A., García, G. J., Pomares, J., Jara, C. A., Delgado Rodríguez, Á., Mateo Agulló, C., Mira Martínez, D., & Pérez Alepuz, J. (2015). *Experiencias sobre el uso de la plataforma Arduino en prácticas de Automatización y Robótica*.

- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *2009 IEEE conference on computer vision and pattern recognition*, 248–255.
- Ketkar, N., & Ketkar, N. (2017). Introduction to Keras. En *Deep Learning with Python* (pp. 97–111). Apress. [https://doi.org/10.1007/978-1-4842-2766-4\\_7](https://doi.org/10.1007/978-1-4842-2766-4_7)
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *ImageNet Classification with Deep Convolutional Neural Networks*. <http://code.google.com/p/cuda-convnet/>
- Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. En *Nature* (Vol. 521, Número 7553, pp. 436–444). Nature Publishing Group. <https://doi.org/10.1038/nature14539>
- Leonardo, M. M., & Faria, F. A. (2019). Um Sistema de Reconhecimento de Espécies de Moscas-das-Frutas. *Anais do XXXVIII Concurso de Trabalhos de Iniciação Científica da SBC*.
- Lu, J., Behbood, V., Hao, P., Zuo, H., Xue, S., & Zhang, G. (2015). Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems*, 80, 14–23. <https://doi.org/10.1016/j.knosys.2015.01.010>
- Massei, M., Yuan, R., Canalis, M. F., Ribotta, G., Druetta, J., & Peretti, G. (2019). La Robótica Educativa: un recurso para potenciar las capacidades científicas-tecnológicas. *XIV Congreso Nacional de Tecnología en Educación y Educación en Tecnología (TE&ET 2019)*, (Universidad Nacional de San Luis, 1 y 2 de julio de 2019).
- Orosco, E. (2019). Clasificación basada en aprendizaje profundo usando cumulantes y biespectro de señales EMG. *IEEE Latin America Transactions*, 17(12), 1946–1953.
- Ortiz de Zúñiga Mingot, I. (2020). *Optimización del sistema de visión artificial de un robot industrial para una aplicación de pick and place*.
- Queiruga, C. A., Banchoff Tzancoff, C. M., Venosa, P., Martín, S. S., Aybar Rosales, V. del C., & Soledad Gomez, I. K. (2021). EscuelasTIC: estrategias para trabajar el pensamiento computacional en la escuela argentina. *XXIII Workshop de Investigadores en Ciencias de la Computación (WICC 2021, Chilecito, La Rioja)*.
- Ramírez, L. G. C., Jiménez, G. S. A., & Carreño, J. M. (2014). *Sensores y actuadores*. Grupo Editorial Patria.
- Rusiñol, M., & Gómez, L. (2018). Avances en clasificación de imágenes en los últimos diez años. *Tábula*, 21, 161–174.
- Seijas, C., Montilla, G., & Frassato, L. (2019). Identificación de especies de roedores usando aprendizaje profundo. *Computación y Sistemas*, 23(1), 257–266.
- Silvero, C., & Escalada, M. (2019). *Escuela de Robótica de Misiones: un modelo de educación disruptiva* (Fundacion). <https://www.fundacionsantillana.com/PDFs/LIBRO-ROBOTICA-WEB-1.pdf>
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Weiss, K., Khoshgoftaar, T. M., & Wang, D. D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1), 1–40. <https://doi.org/10.1186/s40537-016-0043-6>
- Wolyniec Rojas, N. A. (2020). *Detección de lesiones cutáneas en imágenes basado en redes generativas adversarias*.
- Yaranga, C. B., Sanchez, Z., & Rodriguez, R. (2018). Transferencia de aprendizaje mediante redes neuronales convolucionales para el reconocimiento de conductores distraídos. *TECNIA*, 28(2).