



IV Jornadas de Calidad de Software y Agilidad

# JCSA | 2021

12 y 13 de noviembre de 2021



LIBRO DE ACTAS



Libro de Actas de las Cuartas Jornadas de Calidad de Software y Agilidad / Gladys Noemí Dapozo, Emanuel Irrazábal, María de los Ángeles Ferraro, Horacio D. Kuna, Eduardo Zamudio, Alice Rambo, César Acuña, Verónica Bollati y Noelia Pinto ; compilación de Gladys Noemí Dapozo ; Emanuel Agustín Irrazábal. - 1a ed compendiada. - Corrientes : Universidad Nacional del Nordeste. Facultad de Ciencias Exactas, 2021. Libro digital, PDF

Editorial de la Universidad Nacional del Nordeste (EUDENE)

ISBN 978-987-3619-72-4

1. Software. 2. Jornadas. 3. Argentina. I. Dapozo, Gladys Noemí, comp. II. Irrazábal, Emanuel Agustín, comp.

CDD 004.0711

Fecha de catalogación: 03/01/2022

## **Autoridades**

### **Universidad Nacional del Nordeste**

Rectora: Prof. María Delfina Veiravé

Vicerrector: Dr. Mario H. Urbani

### **Facultad de Ciencias Exactas y Naturales y Agrimensura**

Decana: Mgter. María Viviana Godoy Guglielmone

Vicedecano: Dr. Enrique Laffont

### **Universidad Nacional de Misiones**

Rectora: Mgter. Alicia Violeta Bohren

Vicerrector: Ing. Fernando Luis Kramer

### **Facultad de Ciencias Exactas, Químicas y Naturales**

Decano: Dr. Luis Brumobsky

Vicedecano: Dr. Marcelo Marinelli

### **Universidad Tecnológica Nacional**

Rector: Ing. Héctor Aiassa

Vicerrector: Ing. Haroldo Avetta

### **Facultad Regional Resistencia**

Decano: Jorge A. De Pedro

Vicedecano: Dr. Ing. Walter Gustavo Morales

## **IV Jornadas de Calidad de Software y Agilidad**

Este evento es organizado en forma conjunta por las universidades de la región que tienen unidades académicas que ofrecen carreras de Informática, UNNE, UTN y UNaM, a través de un acuerdo específico institucional, con el objetivo de difundir avances significativos en el campo de conocimiento de la Ingeniería de Software, Calidad y Agilidad; y propiciar el encuentro entre las universidades, las empresas y los organismos del Estado para contribuir al desarrollo de la industria del software en la región de influencia de las universidades participantes. Las Jornadas de Calidad de Software y Agilidad (JCSA) se inician en el año 2017, bajo el nombre original de Jornadas de Calidad de Software, en esta edición, las jornadas se consolidaron como un foro regional de referencia, ampliando su espectro para enfatizar también temas relacionados al uso de la agilidad. Además, se propuso la publicación de los trabajos académicos presentados, previa evaluación de pares, que se incorporan en este libro de actas. Durante el evento se realizaron tres talleres vinculados con los temas de la Jornada, se expusieron los artículos y posters académicos aceptados y se presentaron experiencias de la industria del software, con la participación de empresas y organismos del Estado. La conferencia inaugural denominada “Estrategias pruebas de aceptación para Entrega Continua”, estuvo a cargo de Diego Fontdevilla (UNTREF). Las actividades propuestas estuvieron destinadas a ingenieros y licenciados en sistemas, estudiantes y docentes de estas especialidades, profesionales y empresarios del sector del software y servicios informáticos, así como también público interesado en la temática. En total participaron más de 200 personas en las actividades programadas, lo que evidencia el interés que estos temas suscitan en los destinatarios.

## **Comité Organizador**

Mgter. Gladys Noemí Dapozo  
(FaCENA - UNNE)

Dr. Emanuel Irrazábal  
(FaCENA - UNNE)

Lic. María de los Ángeles Ferraro  
(FaCENA - UNNE)

Dr. Horacio D. Kuna  
(FQCEyN - UNaM)

Dr. Eduardo Zamudio  
(FQCEyN - UNaM)

Ing. Alice Rambo  
(FQCEyN - UNaM)

Dr. César Acuña  
(FRRe - UTN)

Dra. Verónica Bollati  
(FRRe - UTN)

Dra. Noelia Pinto  
(FRRe - UTN)

## Comité de Programa

Mgter. Cristina Greiner (GICS - FaCENA - UNNE)	Dr. Diego Godoy (UGD)
Mgter. Gladys Noemí Dapozo (GICS - FaCENA - UNNE)	Ing. Edgardo Belloni (UGD)
Dr. Emanuel Irrazábal (GICS - FaCENA - UNNE)	Mag. Liliana Cuenca Pletsch (CINApTIC - UTN - FRRe)
Dr. Rubén Bernal (GICS - FaCENA - UNNE)	Dr. César J. Acuña (CINApTIC - UTN - FRRe)
Dr. David la Red Martínez (FaCENA - UNNE)	Dra. Verónica Bollati (CINApTIC - UTN - FRRe / CONICET)
Dra. Sonia Mariño (FaCENA - UNNE)	Dra. Noelia Pinto (CINApTIC - UTN - FRRe)
Mgter. M. Viviana Godoy Guglielmono (FaCENA - UNNE)	Esp. Gabriela Tomaselli (CINApTIC - UTN - FRRe)
Mgter. Mónica Tugnarelli (FCAD - UNER)	Ing. Nicolas Tortosa (CINApTIC - UTN - FRRe)
Dra. Gabriela Arévalo (DCyT - UNQ)	Ing. Valeria Sandobal Verón (GIESIN- UTN - FRRe)
Dra. María Fernanda Golobisky (UTN - FRStaFe)	Ing. Germán Gaona (CINApTIC - UTN - FRRe)
Master Ariel Pasini (LIDI - UNLP)	Dr. Horacio Leone (INGAR - UTN - FRSF)
Mgter. Pablo Thomas (LIDI - UNLP)	Dr. Silvio Gonnet (INGAR - UTN - FRSF)
Dr. Fernando Emmanuel Frati (DCByT - UNdeC)	Dr. Gustavo Rossi (LIFIA - UNLP)
Dra. Marcela Genero Bocco (Grupo Alarcos - UCLM)	Dra. Alejandra Garrido (LIFIA - UNLP)
Dr. Jorge Andrés Díaz Pace (ISISTAN - CONICET)	Dr. Andrés Rodríguez (LIFIA - UNLP)
Dr. Nazareno Aguirre (FCEQyN - UNRC / CONICET)	Dr. Marcelo Estayno (UNSAM)
Dra. Nancy Ganz (IIDII-FCEQyN - UNaM)	Dr. Luis Olsina (GIDIS - UNLPam)
Esp. Ing. Alice Rambo (IIDII-FCEQyN - UNaM)	Dra. Luciana Ballejos (CIDISI - UTN - FRSF)
Ing. Selva Nieves Ivaniszyn (FCEQyN - UNaM)	Dra. Luciana Roldan (INGAR - UTN - FRSF)
Lic. Sergio Caballero (FCEQyN - UNaM)	Dra. Milagros Gutierrez (UTN - FRSF)
Lic. Martín Rey (IIDII-FCEQyN - UNaM)	Dra. Mariel Alejandra Ale (CIDISI - UTN - FRSF)
Dr. Eduardo Zamudio (IIDII - FCEQyN - UNaM)	Dra. Elsa Estevez (UNSur / CONICET)
Dr. Horacio Kuna (IIDII - FCEQyN - UNaM)	Dra. Alicia Mon (UNLaM)

## Análisis del uso de frameworks en la estimación de la duración de proyectos web

Yanina Medina, Gladys Dapozo

Facultad de Ciencias Exactas y Naturales y Agrimensura  
Universidad Nacional del Nordeste (UNNE)  
{yanina, gndapozo}@exa.unne.edu.ar

**Resumen**—La estimación de esfuerzo, tiempo y costos es una problemática vigente en el desarrollo de proyectos de software. Este trabajo se orienta a la estimación en el desarrollo de aplicaciones web, dado que la ingeniería web tiene características particulares que requieren de métodos de estimación propios. El objetivo es comparar los resultados de la aplicación de métodos de estimación paramétricos para calcular la duración de proyectos web, y analizar el caso particular de proyectos que utilizan frameworks. Como caso de estudio se utilizó la información obtenida en el desarrollo de un proyecto web realizado por estudiantes universitarios de la asignatura Taller de Programación I, cuyos contenidos se orientan a la programación web. Los resultados obtenidos indican que los métodos más específicos para el desarrollo web y con mayor aporte de información del contexto, generan valores de estimación más precisos, y que la adecuación de los parámetros para reflejar la utilización de frameworks podría mejorar los valores de estimación obtenidos, sin embargo, para asegurar estos resultados los métodos deberían contemplar parámetros específicos para este caso particular.

**Palabras claves**—Estimación software, Desarrollo web, Métodos de estimación paramétricos, Frameworks de desarrollo

### 1. Introducción

En las provincias de Chaco y Corrientes, el sector de Software y Servicios Informáticos (SSI) muestra un desarrollo sostenido, apoyado por los respectivos gobiernos provinciales y/o municipales y las universidades nacionales con carreras de Informática radicadas en la región, contando con polos tecnológicos (Polo IT Chaco y Polo IT Corrientes) que nuclean a las empresas de la zona que buscan, mediante la asociatividad, lograr el crecimiento individual y conjunto; la transferencia de conocimiento, investigación y desarrollo; la búsqueda de alianzas estratégicas con organismos y universidades; y la mejora continua. En este marco, el rol de las universidades es, entre otros, promover la investigación aplicada para incrementar la producción de bienes y servicios que agreguen valor y generen fuentes de trabajo en las áreas estratégicas para el desarrollo, de modo de favorecer la extensión, integración y profundización del proceso de industrialización, el fortalecimiento del mercado interno y la ampliación del comercio exterior [1].

El grupo de Calidad de Software de la Universidad Nacional del Nordeste desarrolla varias líneas vinculadas con la calidad del software, con el propósito de contribuir al desarrollo del sector de SSI de la región, mediante el aporte de métodos y herramientas que incrementen la

calidad de los productos y procesos del desarrollo de software. Una de estas líneas, puntualmente referida a la estimación de proyectos software.

Para las empresas de software una estimación deficiente del esfuerzo y duración que conlleva un proyecto puede ocasionar incumplimiento de plazos, entrega de productos incompletos y pérdida de competitividad. En un estudio realizado en la región NEA [2] se indica que no es habitual que las empresas de software utilicen métodos sistemáticos para estimar duración y costo de los proyectos, la mayoría utiliza las técnicas convencionales de juicio de experto, por analogía y por consenso. Las técnicas paramétricas, son conocidas, pero poco utilizadas. También se observa que la gran mayoría no utiliza datos históricos al momento de realizar la estimación de un proyecto, aspecto que se torna importante para obtener resultados más precisos.

En este trabajo se propone la evaluación de métodos de estimación paramétricos para calcular la duración de proyectos web tomando como caso de estudio el desarrollo de un proyecto realizado por los estudiantes de la asignatura Taller de Programación I de la carrera Licenciatura en Sistemas de Información de la Universidad Nacional del Nordeste (UNNE). Esta información se utiliza para calcular la duración real que se utilizará como referencia para comparar los resultados obtenidos con la aplicación de los métodos seleccionados para este estudio, incorporando también el análisis del uso de frameworks en el desarrollo, y adicionalmente, como información del contexto, el grado de experiencia de los estudiantes.

Para lograr los objetivos propuestos se utilizó una metodología similar a la que se describe en [3], pero considerando que han surgido nuevas técnicas y herramientas para el desarrollo de aplicaciones web, se analiza en particular la utilización de frameworks enfocados en soluciones web. En línea con lo expuesto en [4] que sostienen que los estudios de replicación aumentan la confianza en los resultados anteriores cuando los hallazgos son similares cada vez, y ayudan a madurar el conocimiento abordando aspectos de validez tanto internos como externos.

En la próxima sección se presenta el estado del arte. En la sección 3 se detalla la metodología aplicada. En la sección 4 se presentan los resultados obtenidos. En la sección 5 se presentan las conclusiones. Finalmente se agregó una sección 7 que consiste en un anexo donde se detallan los valores obtenidos en cada uno de los métodos aplicados.

## 2. Estado del arte

### A. Estimación de software

Según McConnell [5] “una estimación es una predicción de cuánto tiempo durará o costará un proyecto”, constituye la base para la planificación de los proyectos. El desarrollo del software requiere de la estimación para controlar y administrar los recursos que se necesitan utilizar antes y durante el proyecto para lograr los resultados esperados.

En [6] se reconoce que, en la gestión y planificación de software, producir una estimación precisa del esfuerzo necesario para completar o mantener un proyecto es de gran importancia y preocupación. Sin embargo, es difícil realizar una estimación realista en la etapa inicial del desarrollo de software, ya que la información disponible en esa etapa suele ser incompleta e incierta. Aunque la construcción de modelos formales de estimación del esfuerzo de software comenzó en los primeros tiempos de la industrialización de la producción de software, el juicio de los expertos sigue siendo la estrategia dominante para la predicción del esfuerzo en la práctica, donde la precisión de la estimación es sensible a la experiencia del profesional y, por lo tanto, es propensa al sesgo.



Carbonera [7] considera que el campo de la estimación del esfuerzo de desarrollo de software tiene un impacto crucial en la presupuestación y la planificación de proyectos en la industria. Para aportar información sobre la literatura actual presenta un estudio de mapeo sistemático sobre la estimación del esfuerzo. Los resultados recopilados indican que los enfoques de estimación de esfuerzo múltiple se utilizan con más frecuencia que uno solo, más del 90% de los estudios tuvieron a los estudiantes como los participantes más comunes en su evaluación empírica. Este artículo informa sobre desafíos que vale la pena investigar, con respecto al uso de la carga cognitiva y la interacción en equipo.

#### *B. Métodos de estimación*

Existen diversos métodos para estimar el esfuerzo de desarrollo, que son clasificados en dos categorías: la primera comprende a los métodos paramétricos, son aquellos en los cuales el proceso de cuantificación del resultado está basado en un proceso mecánico, por ejemplo, la aplicación de una fórmula derivada de los datos históricos. La segunda categoría, comprende a los métodos heurísticos, son aquellos en los cuales la cuantificación del resultado se produce a partir del juicio y/o la experiencia de un experto [8]. En un trabajo previo de los autores se elaboró un cuadro para resumir los principales métodos reportados en la literatura [9].

Los Puntos de Función (PF) constituyen una métrica para establecer el tamaño y complejidad de los sistemas informáticos basada en la cantidad de funcionalidad requerida y entregada a los usuarios. Miden el tamaño lógico o funcional de los proyectos o aplicaciones de software basado en los requerimientos funcionales del usuario. Los proyectos de ingeniería web adoptan normalmente el modelo de proceso ágil. Por este motivo, es frecuente utilizar una medición de puntos de función modificada en conjunto con los pasos de la estimación en proyectos ágiles. Algunos métodos basados en puntos de función fueron utilizados en [3] y que en esta publicación volvieron a validarse. Estos son FP-Lite, Puntos de Casos de Uso (UCP) y WebMo.

La propuesta de FP-Lite deriva del método de análisis de puntos de función definido por el Grupo Internacional de Usuarios de Puntos de Función (IFPUG), simplificando los cálculos para obtener los puntos función -que requerían de mucha documentación que no siempre está disponible-, y dejando como opcional la aplicación del factor de ajuste, dado que el aporte a la precisión que otorga no compensa el esfuerzo que se necesita para calcularlo. Este método puede ser aplicado tanto a proyectos en desarrollo o proyectos de mantenimiento [10].

La metodología UCP se basa en la utilización de casos de uso como dato de entrada para calcular el esfuerzo -en horas-hombre (hh)- necesario para el desarrollo de un proyecto de software [11]. En un intento por dar respuesta a las diferencias entre las estimaciones de desarrollos clásicos y de proyectos web, aparece WebMO, un modelo de estimación orientado a aplicaciones web basado en COCOMO II, propuesto por Reifer [12].

En la literatura se observan nuevas propuestas como ser los Objetos Web Revisados (RWO), siendo esta una actualización del método WO, cuyo objetivo es dar cuenta de los nuevos estilos y tecnologías de desarrollo web. También introduce una clasificación inicial de aplicaciones web de acuerdo con su tamaño, alcance y tecnología, para refinar aún más su estimación de esfuerzo [13]. Otro nuevo enfoque de medición de tamaño es el Factor de Complejidad Web (WCF) para obtener medidas de tamaño para aplicaciones web. WCF tiene cinco factores de medición y es el primer y principal componente para desarrollar un modelo de estimación de esfuerzo para la estimación de esfuerzo web [14]. Según [15] Web objects (WO) es considerada como la primera métrica para estimar tamaño de una aplicación web

desarrollada por Donald, J. Reifer in 2000. WO es una versión extendida de los FP convencionales, obtenidos después de agregar cuatro componentes específicos de web.

### C. Frameworks

Los frameworks de desarrollo tienden a agilizar el desarrollo dado que, utilizan un conjunto de clases que engloban el diseño para resolver un conjunto de problemas comunes existentes en el desarrollo de sistemas. Además, definen elementos imprescindibles para aplicaciones como la automatización de vistas y acceso a datos [16].

La reutilización de componentes, código y automatización de aspectos del desarrollo de sistemas van dirigidos a eliminar el tiempo y el esfuerzo necesarios para construir componentes de software. Este aspecto permite que el desarrollador pueda dedicar más tiempo a tareas y requisitos funcionales, además de centrarse en mantener un código de alta calidad y sostenibilidad [17, 18].

Existen frameworks que dan soporte a diferentes lenguajes de programación para construir aplicaciones web. Según [19], existen los llamados frameworks para PHP, como por ejemplo Kumbia, Wasp, CodeIgniter, Seagull, BlueShoes, Qcodo, Akelos, PhpOpenbiz, Zoop, Ash.MVC, Diy, Wact, Zend Framework, CakePHP, Symfony, Yii, Laravel. En su trabajo, el autor los analizó y comparó con el fin de brindar a los usuarios una idea de cuál elegir para satisfacer sus necesidades, a la hora de realizar tareas con frameworks en lenguaje PHP.

El trabajo que se describe en [20], evalúa varios frameworks de código abierto con el fin de encontrar el más adecuado para el desarrollo de un sistema de apoyo a la toma de decisiones. Los frameworks seleccionados fueron: Rails (Ruby), Django (Python), Grails (Java) y Play (Java).

Según [21], los avances en la tecnología web, en los últimos años han impulsado a los desarrolladores de software a programar aplicaciones móviles responsivas amigables. Las aplicaciones web pueden fácilmente convertirse en aplicaciones móviles y por este motivo, los productos de software se desarrollen continuamente a un ritmo mucho más rápido con nuevas funciones añadidas a diario. Por ejemplo, HTML5 que evoluciona de HTML e incluye nuevos atributos y comportamientos. Aparte de HTML5, los componentes básicos de la mayoría de los navegadores modernos incluyen JavaScript (JS) y hojas de estilo en cascada (CSS3).

Respecto al framework Bootstrap es el marco HTML, CSS y JS más popular para desarrollar proyectos responsive y móviles en la web. Es una biblioteca de código abierto de componentes de UI desarrollada por Twitter. Los componentes están construidos utilizando los principios de diseño web responsivo, lo que hace que esta biblioteca sea extremadamente valiosa para aplicaciones web que necesitan ajustarse automáticamente a su diseño en función de la resolución de la pantalla.

Los autores de [22], luego de que evaluaran la relación que existe entre el uso de un *framework* y la generación de código fuente, concluyen que la mayoría de los métodos de estimación no contemplan la influencia de las herramientas de desarrollo en la generación de código de forma automática; pudiendo dar valores alejados a la realidad.

## 3. Metodología

En este trabajo se presentan los resultados de la aplicación de 3 (tres) métodos de estimación paramétricos para calcular la duración de proyectos web tomando como caso de estudio el desarrollo realizado por los estudiantes de la asignatura Taller de Programación I de la carrera Licenciatura en Sistemas de Información, en cuyo desarrollo se utilizaron frameworks. El objetivo es comparar los resultados de la aplicación de métodos de estimación

paramétricos para calcular la duración de proyectos web, y analizar el caso particular de proyectos que utilizan frameworks.

La metodología se compone de los siguientes pasos:

- Descripción del caso de estudio.
- Cálculo de la duración real del proyecto web desarrollado por los estudiantes
- Estimación de la duración del proyecto con métodos paramétricos
- Comparación de los resultados obtenidos con cada método.
- Ajuste de los parámetros de Webmo para analizar los valores estimados consideran el uso de frameworks y la experiencia de los desarrolladores

#### *D. Descripción del caso de estudio.*

Se denomina caso de estudio al proyecto web que los estudiantes de la asignatura Taller de Programación I de la Licenciatura en Sistemas de Información de la UNNE. Este taller tiene como objetivo que los estudiantes adquieran las técnicas y habilidades necesarias para el desarrollo web, teniendo en cuenta componentes de calidad. Para la aplicación de las herramientas, técnicas y metodologías que conforman los contenidos de la asignatura realizan un proyecto de desarrollo, siguiendo las consignas dadas por los docentes. Con la información aportada por los estudiantes se calcula un valor de “duración real” aproximada con la cual se contrastarán los resultados de duración estimados con los distintos métodos.

El proyecto se desarrolla a largo del cursado (4 meses), se trata de una aplicación web sencilla pero completa, que incluye todos los componentes necesarios: modelado de la aplicación, diseño gráfico y de contenidos, gestor de base de datos, tecnologías de programación en cliente y en servidor. Si bien los estudiantes realizan el desarrollo en forma individual, las características y complejidad del proyecto son las mismas para todos. Esta información se utiliza para asignar los valores de los parámetros de los distintos métodos.

Las consignas dadas a los estudiantes en el cursado 2021, son las siguientes:

- Desarrollar una aplicación orientada al comercio electrónico, respetando los criterios vinculados con la presentación y las funcionalidades requeridas, que fueron previamente definidos por el profesor.
- Realizar la especificación de requerimientos utilizando el estándar de IEEE 830, que incluye el modelado con diagramas de casos de uso.
- Utilizar los frameworks Bootstrap y CodeIgniter, los lenguajes Html5, CSS3 y PHP5; y PhpMy Admin como gestor de datos.
- Realizar la presentación de los resultados del proyecto en 2 fases: la primera incluye la programación en el framework Bootstrap, y la segunda, la programación en el framework CodeIgniter, diseño y conexión de la base de datos.
- Por cada fase, el alumno debe llevar un registro del desarrollo completando un formulario provisto al efecto [23], indicando la tarea realizada y la duración correspondiente.

Además, como información complementaria, se les solicitó indicar el grado de experiencia en programación en un formulario Google docs [23], indicando su nivel de experticia en programación, de acuerdo con las siguientes categorías:

- i. En los últimos 3 años se ha desempeñado como programador en áreas de sistemas o empresas de software.
- ii. No tiene experiencia laboral en programación, pero programa con solvencia, habiendo realizado capacitaciones extracurriculares.
- iii. La experiencia de programación que posee es la de las asignaturas de programación de la carrera, en las que no tuvo problemas.

- iv. Siempre le ha costado la programación y esta es la primera vez que desarrolla un proyecto completo

Al final del plazo indicado, los estudiantes presentaron 15 proyectos de desarrollo individual. Por cada proyecto, se consideró la duración total del mismo, tomando la duración reportada por el propio alumno en el formulario mencionado.

*E. Cálculo de la duración real del proyecto web desarrollado por los estudiantes*

Con la duración consignada por cada estudiante se calculó una duración real como el promedio de las duraciones reportadas por los estudiantes.

*F. Estimación de la duración del proyecto con métodos paramétricos*

Se realizó la estimación de la duración aplicando los métodos FP-Lite, Casos de Uso y Webmo, considerando las características del proyecto realizado, descritas en el apartado A, las herramientas utilizadas y las particularidades del contexto del desarrollo. Se obtuvieron los valores de duración estimados resultantes de cada método.

*G. Comparación de los resultados obtenidos con cada método.*

Los valores estimados se compararon con la duración real calculada.

*H. Ajuste de los parámetros de Webmo para analizar los valores estimados consideran el uso de frameworks y la experiencia de los desarrolladores*

Considerando que el método Webmo es un modelo de estimación orientado a aplicaciones web basado en COCOMO II, propuesto por Reifer [12], considera atributos específicos de este tipo de desarrollo, se propone ajustar los valores de algunos parámetros en los que incidiría el uso de frameworks. Adicionalmente, dado que considera datos propios del contexto, se evalúa también el impacto de la experiencia de los desarrolladores.

## 4. Resultados

A continuación, se presentan los resultados de cada una de las etapas:

*B-Cálculo de la duración real del proyecto*

Al finalizar el plazo previsto, se presentaron 15 proyectos. A partir de los datos obtenidos del formulario destinado al registro de las actividades de cada estudiante, se obtuvo un valor de duración real del desarrollo del proyecto que surge del promedio de las duraciones informadas.

Para la primera fase se estableció una duración promedio de 2 hs por día, durante 14 días. Y para la segunda fase una duración promedio de 4 hs por día, durante 30 días. Considerando la duración medida en meses de 20 días de 5 hs, se obtuvo una duración real de **1,53** meses.

Duración:  $(A+B+C+D) / 5$

A= Promedio de horas en la etapa de análisis

B= Promedio de horas en la etapa de diseño

C= Promedio de horas en la etapa de codificación

D= Promedio de horas en la etapa de prueba

$$(2,26 + 1,73 + 1,73 + 1,93) / 5 = \mathbf{1,53} \text{ meses}$$

*C-Estimación de la duración del proyecto con métodos paramétricos, ajustando sus parámetros para reflejar el uso de frameworks.*

En el Anexo (Ítem 7), se muestran los resultados de la aplicación de cada uno de los métodos considerando las características del proyecto. Cabe mencionar que en todos los casos se consideró el mes de 20 días y 5 hs por día, por la naturaleza de un trabajo académico, en el cual los estudiantes no tienen una dedicación exclusiva al proyecto.

*D- Comparación de los resultados obtenidos con cada método.*

En la tabla 1 se presentan los valores obtenidos y se observa que las estimaciones obtenidas son coherentes con las características de los métodos. Los puntos de casos de uso son un método de estimación temprana con lo cual no incorpora datos que puedan aportar mayor precisión. El método FP-Lite utiliza principalmente las funcionalidades básicas del sistema aportando parámetros en función de las características técnicas del desarrollo y Webmo aporta información específica del contexto de desarrollo de proyectos web.

TABLE I. DURACIONES OBTENIDAS VS DURACIÓN REAL

Duración real	Casos de uso	FP-Lite	Webmo
1,53	3,97	2,7	2,7

*E-Ajuste de los parámetros de Webmo para analizar los valores estimados consideran el uso de frameworks y la experiencia de los desarrolladores*

Para considerar el uso de framework en los multiplicadores de esfuerzo se modificó el valor del parámetro FCIL, relacionado con las facilidades en relación con la disponibilidad de herramientas, al valor **Veryhigh** de acuerdo con las categorías establecidas en [13]. Se obtuvieron los siguientes valores expresados en la tabla 2.

TABLE II. SELECCIÓN DE MULTIPLICADORES DE ESFUERZO

Cost Driver	Ratings				
	Verylow	Low	Nominal	High	Veryhigh
CPLX	0,63	<b>0,85</b>	1	1,3	1,67
PDIF	0,75	<b>0,87</b>	1	1,21	1,41
PERS	1,55	1,35	<b>1</b>	0,75	0,58
PREX	1,35	<b>1,19</b>	1	0,87	0,71
FCIL	1,35	1,13	1	0,85	<b>0,68</b>
SCED	1,35	1,15	<b>1</b>	1,05	1,1
RUSE	1,35	1,15	<b>1</b>	1,25	1,48
TEAM	1,45	<b>1,31</b>	1	0,75	0,62
PEFF	1,35	1,2	<b>1</b>	0,85	0,65

En los Objetos Web se modificó el peso de los parámetros que estarían influenciados por el uso de frameworks, tales como: EQ (archivos de consultas), MMF (archivos multimedia), WBB (bloques de construcción web, como por ejemplo componentes como Shopping cart, botones, logos) y QL (líneas de consultas, script). Los valores obtenidos se muestran en la tabla 3.

TABLE III. RECUENTO DE OBJETOS WEB

Web Objects Predictors	Cantidad x peso	Total
ILF	4x7	28
EI	9x3	27
EQ	2x3	6
MMF	4x3	12
WBB	3x3	9
QL	4x2	8
Total		90

Realizado el cálculo del método Webmo con los parámetros ajustados para reflejar el uso de frameworks, se obtuvieron los resultados que se muestran en la tabla 4.

TABLE IV. ESTIMACIÓN CON WEBMO CONSIDERANDO EL USO DE FRAMEWORKS

<b>Esfuerzo:</b> 203,2 personas-mes
<b>Duración total:</b> 28,5 meses (20 días, 8hs)
<b>Duración del desarrollo:</b> 11,4 meses (40% correspondiente al desarrollo)
<b>Duración del desarrollo ajustado:</b> <b>2,03</b> meses (20 días, 5 hs)

Estimación actual de la duración: **2,03** meses

Dado que en este método se contempla la información del contexto para aportar mayor precisión a la estimación, se evaluó el cálculo considerando la experiencia, tomando como base las categorías en que los estudiantes se ubicaron según su grado de destreza en programación, que se asociaron a los valores del Cost Driver PREX, de la siguiente manera:

Alto: En los últimos 3 años se ha desempeñado como programador en áreas de sistemas o empresas de software.

Nominal: No tiene experiencia laboral en programación, pero programa con solvencia, habiendo realizado capacitaciones extracurriculares.

Bajo: La experiencia de programación que posee es la de las asignaturas de programación de la carrera, en las que no tuvo problemas.

Muy bajo: Siempre le ha costado la programación y esta es la primera vez que desarrolla un proyecto completo

En la tabla 5 se presenta el cambio de valor del multiplicador de esfuerzo PREX, considerando el valor HIGH de 0,87.

TABLE V. SELECCIÓN DE MULTIPLICADORES DE ESFUERZO

Cost Driver	Ratings				
	Verylow	Low	Nominal	High	Veryhigh
CPLX	0,63	0,85	1	1,3	1,67
PDIF	0,75	0,87	1	1,21	1,41
PERS	1,55	1,35	1	0,75	0,58
PREX	1,35	1,19	1	<b>0,87</b>	0,71

Cost Driver	Ratings				
	Verylow	Low	Nominal	High	Veryhigh
FCIL	1,35	1,13	1	0,85	0,68
SCED	1,35	1,15	1	1,05	1,1
RUSE	1,35	1,15	1	1,25	1,48
TEAM	1,45	1,31	1	0,75	0,62
PEFF	1,35	1,2	1	0,85	0,65

Para este nuevo cálculo de estimación, se consideró el recuento de web objects indicados en la tabla 2. Los resultados obtenidos se muestran en la tabla 6.

TABLE VI. ESTIMACIÓN CON WEBMO CONSIDERANDO ALTA EXPERIENCIA DE LOS DESARROLLADORES

<b>Esfuerzo:</b> 148,5 personas-mes
<b>Duración total:</b> 24,3 meses (20 días, 8hs)
<b>Duración del desarrollo:</b> 9,7 meses (40% correspondiente al desarrollo)
<b>Duración del desarrollo ajustado:</b> 1,48 meses (20 días, 5 hs)

Estimación actual de la duración: **1,48** meses

Con los ajustes realizados para reflejar las características del proyecto utilizando frameworks y un nivel alto de experiencia, se puede apreciar que el valor de la estimación mejora, aproximándose a la duración real calculada, tal como se muestra en la tabla 7.

TABLE VII. COMPARACIÓN DE VALORES OBTENIDOS EN WEB CON AJUSTE DE PARAMETROS

Duración real	Webmo	Webmo con Framework	Webmo con experiencia alta
1,53	2,7	2,03	1,48

## 5. Conclusiones

Se aplicaron métodos de estimación paramétricos para calcular la duración de un proyecto *post-mortem*, con el objetivo de evaluar la precisión de la estimación en cada caso. Se utilizaron los métodos basados en Casos de Uso, FP-Lite basados en puntos de función y el método Webmo, este último específico para aplicaciones web. La duración calculada por Webmo y FP-Lite son las que más se aproximan a la duración real y la más alejada es la duración proporcionada por el método de Puntos de Casos de Usos, lo que indica que cuanto más información del proyecto se considere, más precisa será la estimación. De todos modos, la estimación provista por el método basados en Casos de Uso puede ser útil en etapas tempranas del proyecto, apoyando el cálculo del presupuesto del desarrollo.

En el caso particular de Webmo, específico para aplicaciones web, se requiere información de proyectos más avanzado en las etapas de desarrollo, e información histórica y del contexto del desarrollo que aportarán mayor precisión en la estimación.

Cabe destacar que en cada modelo se deben ajustar los parámetros a la situación particular del caso que se analiza, y si bien la selección de los valores puede resultar subjetiva, para las

empresas que desarrollan software utilizar un método paramétrico de estimación representa un buen punto de partida para configurar valores propios en función de la experiencia y del tipo de desarrollo, y generar información histórica que se pueda utilizar en futuros cálculos, mejorando la precisión de la estimación en ese contexto.

Considerando que el desarrollo web actual incorpora nuevas tecnologías, como por ejemplo el uso de frameworks que agilizan el desarrollo, se intentó ajustar los parámetros que establece el método Webmo para reflejar la utilización de esta tecnología, sin embargo, los resultados no muestran una mejora significativa en la estimación utilizando frameworks, lo que sí ocurre ajustando el valor del parámetro de la experiencia. Se concluye que se requiere que el método contemple parámetros específicos para el caso del uso de frameworks.

Como trabajo futuro se propone indagar sobre métodos más actualizados que contemplen específicamente el uso de frameworks en el desarrollo web.

## 6. Referencias

- [1] Cuenca Pletsch, L., Dapozo, G., Greiner, C., & Estayno, M. (2012). Vinculación universidad-empresa orientada a la promoción de la industria del software. Una experiencia de colaboración en la región NEA. Vol. 1 (2012): Revista del Núcleo de Estudios e Investigaciones en Educación Superior del MERCOSUR (versión anterior) ISSN 2313 - 9080
- [2] G. N. Dapozo, C. L. Greiner, E. Irrazábal, Y. Medina, M. Ferraro and B. Lencina, "Características del desarrollo de software en la ciudad de Corrientes", 2015. Disponible en: <http://sedici.unlp.edu.ar/handle/10915/50415>
- [3] G. N. Dapozo, Y. Medina, B. Lencina, G. Pedrozo Petrazzini, "Análisis comparativo de métodos de estimación basados en puntos de función para proyectos web", XX Congreso Argentino de Ciencias de la Computación. (CACIC 2014). - 1a ed. - San Justo: Universidad Nacional de La Matanza, 2014. E-Book. ISBN 978-987-3806-05-6 1. La Matanza, provincia de Buenos Aires, Octubre 2014
- [4] V. Tawosi, F. Sarro, A. Petrozziello, M. Harman "Multi-Objective Software Effort Estimation: A Replication Study". IEEE Transactions on Software Engineering, 2021.
- [5] McConnell, S., Software Estimation: Demystifying the Black Art (Developer Best Practices). Microsoft, 2006.
- [6] F. Sarro, R. Moussa, A. Petrozziello, and M. Harman. "Learning from mistakes: Machine learning enhanced human expert effort estimates". IEEE Transactions on Software Engineering, 2020.
- [7] C.E. Carbonera, K. Farias, V. Bischoff, "Software development effort estimation: a systematic mapping study". IET Software, 2020, vol. 14, no 4, p. 328-344.
- [8] M. Nasir, F. Ahmad, "An Empirical Study to Investigate Software Estimation Trend in Organizations Targeting CMMI", Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science and 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse, 2006.
- [9] G. N. Dapozo, C. L. Greiner, Y. Medina, M. Ferraro, G. Pedrozo Petrazzini, B. Lencina, "Métodos de estimación de software. Un análisis desde un enfoque evolutivo." III Jornadas de Investigación en Ingeniería del NEA y países limítrofes. UTN - Facultad Regional Resistencia. ISBN: 978-950-42-0157-1. 9 y 10 de Junio de 2014.
- [10] Piattini, Medición y Estimación del Software. AlfaOmega Editores, 2008.
- [11] C. A. Remón, P. Thomas, "Análisis de Estimación de Esfuerzo aplicando Puntos de Caso de Uso." XVI Congreso Argentino de Ciencias de la Computación, 2010.
- [12] D. J. Reifer, "Web Development: Estimating Quick-to-Market Software." Software, IEEE Computer Society, November/December 2000, pp. 57-64. 2000
- [13] R. Folgieri, G. Barabino, G. Concas, E. Corona, R. De Lorenzi, M. Marchesi and A. Segni, "A revised web objects method to estimate web application development effort." In Proceedings of the 2nd International Workshop on Emerging Trends in Software Metrics (WETSoM '11). Association for Computing Machinery, New York, NY, USA, 59-64. 2011. DOI:<https://doi.org/10.1145/1985374.1985388>



- [14] S. M. Saif and A. Wahid, "Web complexity factors! A novel approach for predicting size measures for web application development," 2017 International Conference on Inventive Computing and Informatics (ICICI), Coimbatore, 2017, pp. 897-902, doi: 10.1109/ICICI.2017.8365266
- [15] S. M. Saif and A. Wahid, "Web Effort Estimation Using FP and WO: A Critical Study," 2018 Second International Conference on Computing Methodologies and Communication (ICCMC), Erode, 2018, pp. 357-361, doi: 10.1109/ICCMC.2018.8487472.
- [16] R. E. Johnson, "Documenting Frameworks using Patterns. ," p. 14. Accessed on: 2017
- [17] S. H. Kaisler, "Part4 Frameworks con-cepts," in Software Paradigms, I. A John Wiley & Sons, Publication, Ed., 2005.
- [18] D. C. Schmidt, "Introduction to Patterns and Frameworks," prese
- [19] F. Sierra, J. Acosta, J. Ariza y M. Salas, "Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web", Revista Investigación y desarrollo en TIC. Vol. 4 Núm. 2 (2013): Julio – Diciembre.
- [20] L. Teixeira, A.R. Xambre, H. Alvelos, H., N. Filipe & A Ramos, "Selecting an Open-Source framework: a practical case based on software development for sensory analysis", Procedia Computer Science. 64, 1057-1064, 2015.
- [21] F. Shahzad, "Modern and Responsive Mobile-enabled Web Applications", The 12th International Conference on Future Networks and Communications. Procedia Computer Science Volume 110, 2017, Pages 410-415.
- [22] D. Santana Ballate, F. Arteaga Céspedes, V. D. Muñoz Castillo, A. Hernández González, "Evaluación del impacto del uso de un framework en la estimación del esfuerzo de desarrollo del software", Revista Cubana de Ingeniería. Vol. X, N° 1, enero - abril 2019, pp. 69 - 78
- [23] Formulario "Taller de Programación I- Estimación web". Disponible en: <https://bit.ly/3ExgyXK>

## 7. Anexo

### *FP-Lite*

Para la estimación, este método considera el tipo de proyecto a desarrollar. En este caso, dado el contexto específico, los proyectos se consideraron nuevos, multiplataforma, utilizando lenguajes de 4ta. generación. Para el conteo de puntos de función, se analizaron los elementos funcionales requeridos, determinando los siguientes:

- ILF (ficheros lógicos internos): usuarios, productos, ventas, consultas. Total: 4

- ELF (ficheros lógicos externos): no tiene. Total: 0

-EI (entradas externas): agregar productos, agregar usuarios, eliminar productos, eliminar usuarios, modificar usuarios, modificar productos, registrar consultas, eliminar consultas, registrar facturas. Total= 9.

- EQ (consultas externas): consulta de usuarios, consulta de servicios. Total= 2

- EO (salidas externas): no tiene.

La estimación que provee es puntual, pero se recomienda utilizar un intervalo en torno al 20% del valor obtenido. El resultado optimista es el representado por PF-1, el esperado o promedio es PF-2 y el pesimista PF-3.

Dado que el método considera 6 fases para el ciclo de vida (Planificación, especificación, diseño, construcción, pruebas e implantación), en función de las características del proyecto considerado, se tomó en cuenta sólo la fase de Construcción, por tanto, se calculó el esfuerzo de desarrollo como el 40% del Esfuerzo Total, tal como se muestra en la tabla 8.

TABLE VIII. ESTIMACIÓN UTILIZANDO FP-LITE

	Estimación utilizando FP-Lite		
	<i>PF-1 (-20%)</i>	<i>PF-2</i>	<i>PF-3 (+20%)</i>
Total FP sin ajustar	68,8	86	103,2
Esfuerzo total (hs)	965,57	1119,77	1263,88
Duración total	3,92	<b>4,11</b>	4,28
Esfuerzo de desarrollo	386,23	447,91	505,55
Duración (meses 20d x 5h)	2,46	<b>2,72</b>	2,95

La duración se ajustó a las condiciones establecidas (mes de 20 días y día de 5 hs), y se obtuvo el valor de **2,7 meses**.

### *Puntos de Casos de uso*

Para la aplicación de este método se determinaron: Actores: administrador y cliente. Tipo: complejo porque interactúan a través de una interfaz gráfica. Casos de uso: alta, baja, modificación de productos; alta, baja, modificación de usuarios, consulta de usuarios, consulta de productos, registrar consultas de los usuarios, registrar ventas. Tipo: simple; total=10 Factores técnicos.

Utilizando la fórmula

$$TCF = 0,6 + 0,01 \sum_{i=1}^{13} TP_i Po_i, \text{ donde } TP_i \text{ es el peso del factor } i, \text{ y } Po \text{ la ponderación}$$

el valor final de TCF fue de 0,825.

En la tabla 9 se tiene un detalle de la ponderación de cada uno de los factores.

**TABLE IX.** PONDERACIÓN DE CADA FACTOR TÉCNICO DE COMPLEJIDAD

FT	Descripción	Peso	Ponderación	Total
T1	Sistema distribuido	2	0	0
T2	Objetivos de rendimiento	1	2	2
T3	Eficiencia del usuario final	1	3	3
T4	Procesamiento complejo	1	0	0
T5	Reusabilidad	1	3	3
T6	Fácil de instalar	0,5	0	0
T7	Fácil de usar	0,5	3	1,5
T8	Portabilidad	2	3	6
T9	Fácil de cambiar	1	3	3
T10	Uso concurrente	1	0	0
T11	Características de seguridad	1	2	2
T12	Provee acceso a terceros	1	3	3
T13	Formación especial requerida	1	0	0

Se considera que los factores T1, T4, T6, T10 y T13, no son relevantes (ponderación 0) dado que la aplicación no se ha diseñado para que sea distribuida, no hará cálculos matemáticos complejos, el sistema no se instala, no tendrá acceso concurrente y los usuarios no serán expertos (estará orientada para el público en general que desea comprar un artículo o contratar un servicio). Los objetivos de rendimiento y de seguridad no son prioritarios (T2 y T11), por tanto, la ponderación es 2. Los factores valorados con 3 tienen un impacto medio en el proyecto.

Respecto a los factores de entorno, con la siguiente fórmula se obtuvo el valor final de ECF de 0,5.

$$ECF = 1,4 - 0,03 \sum_{i=1}^8 EF_i Po_i, \text{ donde } EF_i \text{ es el peso del factor } i, \text{ y } Po \text{ la ponderación}$$

En la tabla 10 se muestra la valoración que recibió cada uno de los factores:

**TABLE X.** VALORACIÓN DE LOS FACTORES DEL ENTORNO

FE	Descripción	Peso	Ponderación	Total
E1	Familiaridad con el modelo de proyecto utilizado	1,5	3	4,5

FE	Descripción	Peso	Ponderación	Total
E2	Experiencia en la aplicación	0,5	3	1,5
E3	Experiencia en la orientación a objetos	1	4	4
E4	Capacidad del analista líder	0,5	4	2
E5	Motivación	1	5	5
E6	Estabilidad en los requerimientos	2	5	10
E7	Personal de medio tiempo	-1	0	0
E8	Dificultad en el lenguaje de programación	1	3	3

Los factores E1, E2 y E8 se han considerado con un valor promedio debido a la experiencia previa que adquirieron los alumnos en el transcurso de la carrera.

E3 tiene una ponderación relevante porque los alumnos previamente cursaron la asignatura Programación Orientada a Objetos y E4 porque el analista líder es el propio profesor.

E5 tiene la más alta ponderación debido a que el proyecto propone libertad para aplicar las soluciones, representa un desafío e implica la aprobación de la materia.

Igual que E6 que justifica su ponderación porque no habrá cambios en los requerimientos planteados al inicio.

E7 es 0 porque el proyecto no cuenta con personal de medio tiempo.

Se ha tomado un factor de productividad (FP) de 17 al ser casos de usos simples. El cálculo final se presenta en la tabla 11.

TABLE XI. ESTIMACIÓN UTILIZANDO CASOS DE USO

<p>Puntos Caso de Uso sin ajustar (UUCP – Unadjusted Use Case Points)</p> $UUCP = \sum_{i=1}^n A_i P_i + \sum_{i=1}^m CU_i P_i$ <p>donde n es el nº de actores, m el nº de CU y P<sub>i</sub> el peso</p> <p>UUCP = Actores * Peso + Casos de uso * Peso = 2 * 3 + 10 * 5 = 56</p>
<p>Puntos Caso de Uso ajustados (UCP)</p> $TCF = 0,6 + 0,01 \sum_{i=1}^{13} TF_i P_{o_i}$ <p>donde TF<sub>i</sub> es el peso del factor i, y P<sub>o</sub> la ponderación</p> $ECF = 1,4 - 0,03 \sum_{i=1}^8 EF_i P_{o_i}$ <p>donde EF<sub>i</sub> es el peso del factor i, y P<sub>o</sub> la ponderación</p> <p>UCP = TFC * ECF * UUCP = 0,835 * 0,5 * 56 = 23,38</p>
<p><b>Esfuerzo</b> = UCP * FP = 23,38 * 17 = 397,46 hh</p>
<p><b>Duración</b> = Esfuerzo / horas por día / días por meses = 397,46 / 5 / 20 = <b>3,97 meses</b></p>

Estimación actual de la duración: **3,97 meses**.

### WebMO

Para este método de estimación se ha determinado que los proyectos son de tipo comercio electrónico, por lo que los valores de las constantes de esfuerzo proporcionadas por WebMo son: A=2,3; B=2; P1=1,03 y P2=0,5.

Como los alumnos programaron en diversos lenguajes (PHP, HTML5, CSS3, SQL) se consideró el default para los lenguajes de 4GL = 20 para la constante de lenguaje.

Para establecer el valor de los multiplicadores de esfuerzo, se tomaron los valores establecidos en la tabla Cost Driver que define el método [13].

TABLE XII. SELECCIÓN DE MULTIPLICADORES DE ESFUERZO

Cost Driver	Ratings				
	Verylow	Low	Nominal	High	Veryhigh
CPLX	0,63	<b>0,85</b>	1	1,3	1,67
PDIF	0,75	<b>0,87</b>	1	1,21	1,41
PERS	1,55	1,35	<b>1</b>	0,75	0,58
PREX	1,35	<b>1,19</b>	1	0,87	0,71
FCIL	1,35	1,13	1	<b>0,85</b>	0,68
SCED	1,35	1,15	<b>1</b>	1,05	1,1
RUSE	1,35	1,15	<b>1</b>	1,25	1,48
TEAM	1,45	<b>1,31</b>	1	0,75	0,62
PEFF	1,35	1,2	<b>1</b>	0,85	0,65

El recuento de objetos web (WO) se calculó seleccionando los valores que corresponden a cada aspecto del proyecto. Los valores resultantes se muestran en la tabla 13.

Tamaño = WO \* constante del lenguaje / 1000

Tamaño = 98 \* 20 / 1000 = 1,96 (Miles de líneas de código)

TABLE XIII. RECUENTO DE OBJETOS WEB

Web Objects Predictors	Cantidad x peso	Total
ILF	4x7	28
EI	9x3	27
EQ	2x3	6
MMF	4x4	16
WBB	3x3	9
QL	4x3	12
Total	98	98

Para calcular el esfuerzo y la duración, se utilizan las siguientes fórmulas:

**Esfuerzo** = A ([cd;] (tamaño)<sup>P1</sup>

**Duración** = B (esfuerzo)<sup>P2</sup>

Donde:

- A, B son constantes multiplicativas de esfuerzo y P1, P2 exponentes, que se determinan en función del tipo de proyecto y del tamaño. Ver tabla 14.

- Cdi: Manejadores de costos.
- Tamaño: líneas de código fuente, expresado en miles. Se obtiene multiplicando el total de Web Objects por una constante asociada al lenguaje de programación.

**TABLE XIV. CONSTANTES DE ACUERDO AL TIPO DE PROYECTO**

Parámetros / Exponentes	A	B	P1	P2
Basado en el comercio electrónico	2,3	2	1,05	*
Aplicación financiera	2,7	2,2	1,05	*
Aplicación de B2B (negocio a negocio)	2	1,5	1	*
Aplicación basada en utilitarios informativos	2,1	2	1	*

(\*) 0,5 para menos de 300 puntos de objeto o 0,32 para más 300.

Para calcular el producto de los manejadores de costos (cdi) se seleccionan los valores (que se detallan en [13]), asociados a distintos aspectos vinculados con el proyecto:

- CPLX: Complejidad y confiabilidad del producto
- PDIF: Dificultad de la plataforma
- PERS: Capacidades del personal
- PREX: Experiencia del personal
- FCIL: Facilidades
- SCED: Restricciones de horarios
- RUSE: Grado de reutilización planeado
- TEAM: Trabajo en equipo
- PEFF: Eficiencia del proceso

Los Web Objects se calculan a partir de los siguiente predictores:

- ILF: Archivos lógicos internos.
  - EIF: Archivos externos
  - EI: Entradas externas
  - EO: Salidas externas
  - EQ: Consultas externas
  - MmF: Archivos multimedia
  - WBB: Constructor de bloques web
  - Scripts: número de macros, contenedores, etc
  - Links: número de referencias
- asignándole a cada uno un peso (bajo, medio, alto) según corresponda.

En la tabla 15 se muestran los valores resultantes de la estimación de esfuerzo y duración.

**TABLE XV. ESTIMACIÓN CON WEBMO**

<b>Esfuerzo:</b> 277,7 personas-mes
<b>Duración total:</b> 33,3 meses (20 días, 8hs)
<b>Duración del desarrollo:</b> 13,3 meses (40% correspondiente al desarrollo)
<b>Duración del desarrollo ajustado:</b> 2,7 meses (20 días, 5 hs)

Estimación actual de la duración: 2,7 meses.