



XX Workshop de Investigadores en Ciencias de la Computación
26 y 27 de abril de 2018 - Corrientes - Argentina

LIBRO DE ACTAS

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

Red de Universidades con Carreras de Informática (RedUNCI)



Dapozo, Gladys

XX Workshop de Investigadores en Ciencias de la Computación - WICC 2018 : libro de actas / Gladys Dapozo ; Patricia Pesado ; compilado por Gladys Dapozo ; Emanuel Irrazabal. - 1a ed compendiada. - Corrientes : Universidad Nacional del Nordeste. Facultad de Ciencias Exactas, 2018.

Libro digital, PDF

Archivo Digital: descarga y online

ISBN 978-987-3619-27-4

1. Bases de Datos. 2. Minería de Datos. 3. Ingeniería de Software. I. Pesado, Patricia II. Dapozo, Gladys , comp. III. Irrazabal, Emanuel , comp. IV. Título.

CDD 004.071

[Ingeniería de software para sistemas embebidos, requisitos en PYMEs y testing continuo

Irrazábal, Emanuel¹; Mascheroni, Maximiliano Agustín¹; Alonso, José Manuel¹, Vier, Sergio², Pereyra Coimbra, Rodrigo²

1: Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura. Universidad Nacional del Nordeste

2: Departamento de Informática, Facultad de Ciencias Exactas Químicas y Naturales, Universidad Nacional de Misiones.

{eirrazabal, mascheroni}@exa.unne.edu.ar, alonso.651@gmail.com,
joroimbra@gmail.com sergiovier@gmail.com

Resumen

Esta línea de investigación aborda temas de ingeniería del software en sistemas tradicionales y su traslado a sistemas embebidos. Se espera la aplicación de la tecnología de desarrollo software en sistemas embebidos para el ámbito regional orientado a las entidades de ciencia y tecnología. En este sentido se está trabajando, por un lado, la construcción de un modelo de procedimientos para la gestión de requerimientos en entidades agrícolas con una cultura organizacional jerárquica, específicamente las entidades yerbateras del nordeste argentino. Y por otro lado en el desarrollo de sistemas embebidos de riego.

Finalmente se está trabajando en el desarrollo de procedimientos para la formalización del testing continuo en la disciplina de entrega continua lo cual facilitará los ensayos en los desarrollos ágiles.

Palabras clave: Ingeniería de software, requisitos, entrega continua, sistemas embebidos, riego

Contexto

La línea de Investigación y Desarrollo presentada en este trabajo corresponde al proyecto PI-F17-2017 “Análisis e implementación de tecnologías emergentes en sistemas computacionales de aplicación regional.”, acreditado por la Secretaría de Ciencia y Técnica de la Universidad Nacional del Nordeste (UNNE) para el periodo 2018-2021.

Asimismo, parte de la línea de investigación es realizada en el marco de la tesis de los maestrandos Sergio Fabián Vier y Rodrigo Pereyra Coimbra pertenecientes a la Maestría de Tecnologías de la Información Rs. 764/14 CS UNNE.

Introducción

Ingeniería de software para sistemas embebidos

Un sistema embebido es aquel sistema basado por lo general en un microprocesador, sensores y actuadores diseñado para realizar funciones dedicadas [1]. Y han cobrado gran importancia desde el punto de vista de los sistemas de información con el uso de plataformas tipo Arduino para el desarrollo rápido de prototipos [2].

En este sentido, el foco del grupo de investigación estará puesto en el desarrollo

de soluciones para entidades regionales, en particular para la agricultura y los grupos de investigación de la universidad. La actividad agrícola es una de las principales fuentes de producción de alimentos en el ámbito mundial. Las nuevas tecnologías de la computación y la comunicación ofrecen soluciones que pueden adaptarse a las necesidades de la agricultura para una producción más eficiente en el uso de recursos y por lo tanto más sustentable económica, ambiental y socialmente. Entre ellas se encuentran los desarrollos relacionados con la automatización del riego. Desde hace varios años se han implementado sistemas de riego que permiten un uso más racional del agua con niveles de desarrollo óptimos, en los cultivos donde se implementen [3][4][5][6][7]. Para los diferentes cultivos existen requerimientos particulares con los que lograr un óptimo desarrollo y productividad. Cada fase del desarrollo del cultivo requiere un mínimo de acumulación de temperatura, así como necesidades hídricas y nutricionales que varían a lo largo de cada fase. Esto hace indispensable tener un control asistido de estos parámetros [8]. Asimismo, el Instituto Nacional de Tecnología Agropecuaria (INTA) desarrolla diferentes líneas de investigación para mejorar el desarrollo de los cultivos [9]. Y una de sus necesidades es asegurar el control de las variables ambientales en los laboratorios e invernáculos. Por lo tanto, la inclusión de los sistemas de riego automatizados permite el uso eficiente del agua, es capaz de mejorar el rendimiento de los cultivos y es un insumo para el desarrollo de actividades de investigación agropecuaria.

Ahora bien, los esquemas de riego estáticos tradicionales no se ajustan adecuadamente a las necesidades hídricas de los cultivos en producción o de los ensayos en el ámbito de investigación [10]. El cambio en las temperaturas y las precipitaciones durante el día requieren, por lo tanto, un manejo hídrico no lineal.

De la misma manera, el desarrollo de tecnologías sustitutas en los laboratorios de investigación son una práctica corriente en los países en vías de desarrollo, por su alto costo y las barreras de mantenimiento [11][12]; esto especialmente en el interior del país. Asimismo, permite la personalización del equipamiento y la inclusión tecnologías como el diseño 3D [13] o los sistemas embebidos para la monitorización y el control de los procesos del laboratorio [14].

Ingeniería de Software para Requisitos y Entrega Continua

Tal y como lo describe la segunda ley de Lehman [15], el software necesita ser cada vez más complejo para satisfacer las necesidades de los usuarios. Esto, sumado a la importancia de detectar errores en la etapa de captura de requerimientos por sus costos [16], hace a la validación de los requerimientos una etapa crítica en el desarrollo de las aplicaciones [17].

A su vez, la mayoría de las Empresas Yerbateras que se encuentran en la provincia de Misiones tienen una estructura jerárquica tradicional, donde se identifican necesidades para gestión y seguimiento de tareas relacionadas a la Ingeniería de Software [18]. Estas empresas en su mayoría cuentan con un directorio que toma las decisiones, el cual conoce el negocio, pero no cuenta con el tiempo o la heurística suficientes para manifestar sus necesidades.

En este sentido, los desarrollos de metodologías ágiles representan un avance en la manera de construir sistemas; usando métodos, como, por ejemplo: el prototipado, las historias de usuario y los casos de uso. Esto hace posible la entrega temprana de valor, la respuesta rápida a los cambios y la colaboración constante del equipo de trabajo con los clientes [19][20].

Entonces, haciendo foco en la gestión de los requerimientos, es posible adaptar técnicas ágiles las cuales tendrán un fuerte impacto positivo [21][22], especialmente

en PYMEs con problemas al momento de la gestión de sus requerimientos.

Las herramientas y prácticas genéricas deben ser seleccionadas y adaptadas para la empresa, tamaño y dominio específico. Y así, reducir los problemas causados por el no alineamiento tales como requisitos implementados incorrectamente, retrasos y esfuerzos desperdiciados [25].

El Sistema de Información Universitaria (SIU) [26] inicia en 1996 desarrollando sistemas informáticos para la gestión de diversas áreas en instituciones que componen el sistema universitario nacional. El SIU cuenta con 54 proyectos de software activos [27]. Uno de los proyectos recientes y actualmente en desarrollo es el SIU-Araí [28]. Se trata de una plataforma integradora de servicios, donde cada uno de los sistemas desarrollados por el SIU es considerado un módulo. De esta forma, la plataforma SIU-Araí es concebida como un componente central en las soluciones que proporciona el SIU hacia las instituciones del sistema universitario argentino.

A la par, la industria del software está moviéndose hacia la adopción de un paradigma en el cual la funcionalidad del software está en continua evolución [29]. Y el SIU busca llevar adelante el desarrollo de las soluciones software con un enfoque evolutivo.

Esto es cubierto por el enfoque de la ingeniería del software conocido como Entrega Continua de Software, en inglés Continuous Delivery (CD). Este es un enfoque en el cual los equipos mantienen la producción de software en ciclos cortos de tiempo, asegurando que el producto pueda ser lanzado de manera fiable en cualquier momento [30]. El objetivo es poder lanzar a producción un producto software libre de defectos “*con solo apretar un botón*” [31].

Uno de los principales desafíos de este enfoque es la calidad del producto software resultante. Ésta puede disminuir, dado que, al realizarse los despliegues del sistema con mayor frecuencia, aparecen más

defectos en el producto. Por tanto, es esencial desarrollar un enfoque de priorización de los diferentes aspectos en la calidad del producto software, teniendo en cuenta la forma de trabajo actual de las empresas de desarrollo software.

Líneas de investigación y desarrollo

En la línea de Ingeniería de Software para Sistemas Embebidos se propone:

- Desarrollar soluciones de sistemas embebidos con microcontroladores que solucionen problemas regionales y apoyen a los grupos de investigación locales.

En la línea de Ingeniería de Software para los Requisitos y la Entrega Continua:

- Desarrollar un modelo de procedimientos para la gestión de requerimientos en PYMEs yerbateras a partir de técnicas ágiles.
- Diseñar un modelo de pruebas de software mediante la combinación de las técnicas existentes para su uso en el SIU.
- Desarrollar una herramienta que permita la ejecución de pruebas de manera continua basadas en el modelo diseñado.

Resultados obtenidos

El grupo de investigación es de reciente formación, por lo cual los resultados son preliminares y, en parte, se enumeran antecedentes llevados adelante en el marco de otros grupos de trabajo. A continuación se indican:

En la línea de Ingeniería de Software para Sistemas Embebidos:

- Se realizaron dos presentaciones a congresos especificando el desarrollo de un Planificador Embebido para la Gestión de Riego Automatizado (PEGRA). En el primer artículo se describe el diseño y la construcción del prototipo [32]. En el segundo artículo se

detalla el diseño con componentes de uso específico y mejoras al diseño [33].

En la línea de Ingeniería de Software para la Entrega Continua:

- En [32] se realizó una revisión sistemática de la literatura. En ella, se buscaron propuestas, técnicas, enfoques, métodos, herramientas y otro tipo de soluciones para afrontar los problemas de pruebas en entornos de desarrollo continuo. También se validó que las pruebas continuas son un elemento faltante en la entrega continua, y se analizaron las diferentes definiciones de la misma y los diferentes niveles y etapas de pruebas. Por último, se detectaron nuevos problemas aún no resueltos relacionados a las pruebas continuas.
- En [33] se llevó a cabo una revisión sistemática con el objetivo de identificar, analizar y sintetizar las técnicas, herramientas y desafíos encontrados en la literatura sobre pruebas de compatibilidad web. Los resultados indicaron que la técnica de prueba más elegida es el análisis visual. Asimismo, el principal desafío detectado es la detección de elementos variables.

Formación de recursos humanos

En el Grupo de Investigación en Innovación en Software y Sistemas Computacionales (GISSC) están involucrados 4 docentes investigadores, 1 becario de investigación de pregrado, 1 tesista de doctorado y 3 tesistas de maestría. Cinco alumnos de la carrera están realizando sus proyectos finales vinculado a estos temas.

Referencias

- [1] Heath, Steve. *Embedded systems design*. Elsevier, 2002.
- [2] Jamieson, Peter. "Arduino for teaching embedded systems. are computer scientists and engineering educators missing the boat?." Proc. FECS 2010 (2010): 289-294.
- [3] Martínez, Germán Arturo López, Moreno, Flor García y Fierro, Juan Bedoya. Modelo a escala de un sistema de riego automatizado, alimentado con energía solar fotovoltaica: nueva perspectiva para el desarrollo agroindustrial colombiano. 2014, Revista Tecnura, Vol. 17, págs. 33-47.
- [4] Castro Silva, Juan Antonio. Sistema de riego autónomo basado en la internet de las cosas. Neiva, Colombia: s.n., 2016.
- [5] Escalas Rodríguez, Gabriel. Diseño y desarrollo de un prototipo de riego automático controlado con Raspberry Pi y Arduino. Barcelona, España: s.n., 2014.
- [6] Escamilla Martínez, Fernando. Automatización y telecontrol de sistema de riego. Gandía, España: s.n., 2016.
- [7] Hernandez Garcia, C. Itzel. Diseño de un sistema automatizado de riego para el cultivo de tomate y pimiento bajo invernadero. Xalapa Enríquez, México: Diciembre de 2014.
- [8] An Intelligent Smart Irrigation System Using WSN and GPRS Module. Manimaran , P. . 6, 2016, International Journal of Applied Engineering Research, Vol. 11, págs. 3987-3992.
- [9] Instituto Nacional de Tecnología Agropecuaria. INTA. [En línea] 2013. [Citado el: 12 de Junio de 2017.]
- [10] Pereira, Luis Santos, y otros, y otros. Riego localizado. El riego y sus tecnologías. Albacete, España : Europa-América, 2010, págs. 229 - 231.
- [11] Pearce, Joshua M. Open-source lab: how to build your own hardware and reduce research costs. Newnes, 2013.
- [12] Lazalde, Alan, Jenny Torres, and David Vila-Viñas. "Hardware: ecosistemas de innovación y producción basados en hardware libre." Buen Conocer-FLOK Society. Modelos sostenibles y políticas públicas para una economía social del conocimiento común y abierto en el Ecuador. Asociación aLabs, 2015. 619-652.
- [13] Baden, Tom, et al. "Open Labware: 3-D printing your own lab equipment." PLoS biology 13.3 (2015): e1002086.
- [14] Pearce, Joshua M. "Building research equipment with free, open-source hardware." Science 337.6100 (2012): 1303-1304.

- [15] M. Lehman, "On Understanding Laws, Evolution and Conservation in the Large Program Life Cycle", *J. of Sys. and Software*, vol. 13, pp. 213-221, 1980.
- [16] S. Maalem and N. Zarour, "Challenge of validation in requirements engineering", *Journal of Innovation in Digital Ecosystems*, vol. 3, no.1, pp. 15-21, 2016.
- [17] G. Kotonya and I. Sommerville, *Requirements engineering*. Chichester: John Wiley & Sons, 1998.
- [18] Plan Estratégico para el Sector Yerbatero – Yerba Mate Argentina", *Yerbamateargentina.org.ar*, 2016.
- [19] A. Sillitti and G. Succi, *Requirements Engineering for Agile Methods*. In: Aurum A., Wohlin C. (eds) *Engineering and Managing Software Requirements*. Berlin, Heidelberg: Springer, 2005.
- [20] I. Inayat, S. Salim, S. Marczak, M. Daneva and S. Shamshirband, "A systematic literature review on agile requirements engineering practices and challenges", *Computers in Human Behavior*, 2014.
- [21] S. Dragicevic, S. Celar and L. Novak, "Use of Method for Elicitation, Documentation, and Validation of Software User Requirements (MEDoV) in Agile Software Development Projects", 2014 Sixth International Conference on Computational Intelligence, Communication Systems and Networks, 2014.
- [22] One, V. "State of agile development survey results", 2017.
- [23] E. Bjarnason, P. Runeson, M. Borg, M. Unterkalmsteiner, E. Engström, B. Regnell, G. Sabaliauskaite, A. Loconsole, T. Gorschek and R. Feldt, "Challenges and practices in aligning requirements with verification and validation: a case study of six companies", *Empirical Software Engineering*, vol. 19, no. 6, pp. 1809-1855, 2013.
- [24] «Sistema de Información Universitario», ¿Qué hacemos? [En línea]. Disponible en: <http://www.siu.edu.ar/que-hacemos/>. [Accedido: 24-jul-2017].
- [25] «Sistema de Información Universitario», *Nuestra gestión en números*. [En línea]. Disponible en: <http://www.siu.edu.ar/nuestra-gestion-en-numeros/>. [Accedido: 24-jul-2017].
- [26] «Documentación - SIU-Arai». [En línea]. Disponible en: <http://documentacion.siu.edu.ar/wiki/SIU-Arai>. [Accedido: 25-jul-2017].
- [27] J. Järvinen, T. Huomo, T. Mikkonen, y P. Tyrväinen, «From agile software development to mercury business», en *International Conference of Software Business*, 2014, pp. 58–71.
- [28] L. Chen, "Continuous Delivery: Huge Benefits, but Challenges Too" in *IEEE Software* 03/2015. V. 32(2).
- [29] J. Humble and D. Farley. "Continuous delivery: reliable software releases through build, test, and deployment automation", 1st ed. Boston, US: Pearson Education, 2010.
- [30] Alonso, JM; Ferrari, H; Sambrana, I; Irrazábal, E. Emanuel Irrazábal. Desarrollo de un Planificador embebido para la gestión de riego automático aplicado a invernáculos. Implementación en INTA El Sombrero. XXIII Congreso Argentino de Ciencias de la Computación CACIC 2017. ISBN: 978-987-4251-43-5 La Plata (Argentina). Fecha: Octubre de 2017
- [31] Alonso, JM; Ferrari, H; Sambrana, I; Irrazábal, E. Desarrollo de un Sistema de Gestión de Riego Automático Basado en la Humedad del Suelo. 5to Congreso Nacional de Ingeniería Informática / Sistemas de Información Aplicaciones Informáticas y de Sistemas de Información. Libro de Actas CONAISSI 2017. Santa Fe (Argentina). Fecha: 2 y 3 de Noviembre de 2017
- [32] Mascheroni, M.A., Irrazábal, E. (in press). Continuous Testing and solutions for testing problems in Continuous Delivery: A Systematic Literature Review. *Computación y Sistemas*.
- [33] Sabaren, L; Mascheroni, M.A.; Greiner, C; Irrazábal, E. Una Revisión Sistemática de la Literatura en Pruebas de Compatibilidad Web XXIII Congreso Argentino de Ciencias de la Computación CACIC 2017. ISBN: 978-987-4251-43-5. La Plata (Argentina). Fecha: Octubre de 2017