

APLICACIÓN DE METODOLOGÍAS EMERGENTES PARA CONTRIBUIR CON LA CALIDAD DEL SOFTWARE

Irrazabal, Emanuel Dapozo; Gladys N.; Greiner, Cristina;
Company, Ana M.; Mascheroni, Agustín,

Facultad de Ciencias Exactas y Naturales y Agrimensura.
Universidad Nacional del Nordeste
{eirrazabal, gndapozo, cgreiner, eirrazabal, mascheroni }@exa.unne.edu.ar,
anamacom@gmail.com

RESUMEN

Este es el segundo año del proyecto F018-2017; una continuación de los proyectos F07-2009 y F10-2013, ambos enfocados en modelos, métodos y herramientas para la calidad del software. Este tercer proyecto aborda los temas emergentes en el área de la calidad de software, en particular, aspectos vinculados con la gestión de los procesos administrativos mediante herramientas software y en la entrega continua de productos software. El equipo de trabajo busca promover y/o generar métodos y herramientas que contribuyan a mejorar la calidad del proceso y del producto software.

En particular, se está trabajando en el desarrollo de procedimientos para la gestión de las actividades de prueba continua dentro de la disciplina de entrega continua. Esto ha incluido el desarrollo de revisiones sistemáticas de la literatura y el análisis comparado de las herramientas actuales de entrega continua y la evaluación de sus características. Finalmente también se atiende la problemática de la eficiencia de los sistemas en organismos públicos, mediante el enfoque de procesos de negocios y el desarrollo rápido de aplicaciones.

Palabras clave: calidad de software, gestión de procesos de negocio, entrega continua.

CONTEXTO

Las líneas de Investigación y Desarrollo presentada en este trabajo corresponden al proyecto PI-17F018 “Metodologías y herramientas emergentes para contribuir con la calidad del software”, acreditado por la Secretaría de Ciencia y Técnica de la Universidad Nacional del Nordeste (UNNE) para el periodo 2018-2021.

1. INTRODUCCIÓN

Las características de un producto software necesitan ser cada vez más complejas para satisfacer a los usuarios [1]. Esto genera avances constantes que se traducen en una gran cantidad de nuevas técnicas y tecnologías relacionadas con el desarrollo de software [2]. Pero el desafío sigue siendo llevar adelante un proceso de desarrollo que termine en un plazo acordado con unos atributos de calidad esperados, como, por ejemplo la mantenibilidad o la usabilidad.

En este sentido, la calidad del software está formada por diferentes factores y apoyados por herramientas que automatizan su medición. Por lo tanto, las nuevas tecnologías y técnicas emergentes muchas veces no tienen el conjunto de herramientas necesarias para la medición y mejora de la calidad del producto software resultante [3]. Por ello la necesidad de investigar en el uso y adecuación de las tecnologías emergentes incorporando aspectos de calidad del software.

Autores como Pressman [4], McCall [5], y estándares tales como la ISO 25010 [6] han tratado de determinar y categorizar los factores que afectan a la calidad del software. Y en general se concluye que la calidad interna del producto software está directamente relacionada con diferentes atributos cuantificables a partir de su código fuente. Existen diferentes estudios que relacionan estos atributos con herramientas y métricas [7].

Asimismo el análisis del código fuente a partir de estas métricas puede ser tenido en cuenta para describir la complejidad del producto software y la probabilidad de encontrar errores [8]. Pero en las tecnologías emergentes esto no está del todo desarrollado. Así por ejemplo, no se encuentran herramientas o técnicas de pruebas software para productos software nativos móviles [3].

Siguiendo con este mismo enfoque y con el objetivo de lograr una mayor satisfacción del cliente, es crítico para las organizaciones entregar productos de calidad de manera aún más rápida. Esto dio lugar al surgimiento de un nuevo enfoque denominado “Entrega Continua de Software”, más conocido en inglés como Continuous Delivery (CD). En este enfoque 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 [10] [11].

Nuevamente, uno de los principales problemas de estos enfoques estaría en la calidad del producto software. Ésta puede disminuir, dado que, al realizarse los despliegues del sistema con mayor frecuencia, aparecen más defectos en el producto [11]. 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.

Para adoptar este enfoque, se utiliza el concepto de Tubería de Despliegue (DP -

Deployment Pipeline), un estándar para automatizar el proceso de CD [12].

Desde el enfoque de los procesos de construcción también existen desafíos, no solamente desde el sector privado sin también en el sector público. La gestión de procesos de negocio se basa en la idea de que cada producto es el resultado de un conjunto de actividades que se realizan a fin de obtener dicho producto. Por este motivo, la correcta y eficiente gestión de los procesos de negocio es un aspecto importante para la productividad de toda organización, ya que permite identificar las tareas, el orden de ejecución de las mismas y las personas responsables de realizarlas [13].

Por lo tanto, las líneas de trabajo del proyecto tienen que ver con características de calidad de calidad del producto software y del proceso de desarrollo.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

A continuación, se describen las siguientes líneas de investigación y desarrollo:

Entrega continua

Se está realizando el desarrollo de un conjunto de procedimientos para la gestión de las pruebas continuas teniendo en cuenta los principales desafíos de esta disciplina y los enfoques más actuales para agilizar el proceso de pruebas. En esta línea estamos especialmente orientados a las pruebas de tipo API Rest, las pruebas de visualización en navegadores y las pruebas de tipo flaky. Como primera medida se han realizado revisiones sistemáticas de la literatura y como segundo paso el desarrollo de encuestas a grandes empresas mundiales.

El paso siguiente es la creación de un primer modelo formado por distintas etapas de pruebas basándose en la Tubería de Despliegue. Para validar el modelo, se procederá con el desarrollo de un marco de

trabajo que implemente estas pruebas. Se hará una serie de experimentos con este marco para comparar la velocidad de ejecución de las pruebas contra las convencionales. Y, finalmente, se propone implementar el modelo en un ambiente real de desarrollo, para analizar los resultados obtenidos.

Gestión de procesos de negocios en el sector público

Los proyectos de desarrollo software en la Administración Pública requieren cada vez mayor eficiencia, respecto a la optimización en el uso de los recursos involucrados; pero además deben lograr sumar apoyo en la estructura del estado y su cultura organizacional.

En este sentido, la modelización rápida de prototipos para la automatización de las áreas administrativas es fundamental. Así, por ejemplo, una de las herramientas más utilizadas es Bonita [14], una plataforma que integra tecnologías para el desarrollo de procesos de negocio. Esta herramienta además permite realizar entregas de aplicaciones operativas con rapidez y dispone de herramientas que posibilitan la mejora continua.

Incluye herramientas de modelización que siguen la notación BPMN 2.0 y contiene un motor de ejecución de procesos, o, por ejemplo, funcionalidades de integración para conectar las aplicaciones con servicios de terceros. Esto favorecerá la usabilidad del proceso [15].

En esta línea de investigación se busca realizar un modelo de trabajo para mejorar la gestión del proceso administrativo de las instituciones públicas. Por un lado, el modelado mediante la notación BPM y por otro lado el desarrollo de prototipos rápidos mediante la herramienta antes descrita.

3. RESULTADOS OBTENIDOS/ESPERADOS

En el marco de este proyecto y respecto de la línea de gestión de procesos se lograron los siguientes objetivos:

1. Relevamiento de información relevante sobre los conceptos involucrados, procesos de negocios y usabilidad

En primer lugar, se realizó una revisión sistemática de la literatura (RSL) con el objetivo de sintetizar la literatura existente sobre procesos de desarrollo de software basados en procesos de negocio y su vinculación con el diseño y la usabilidad de las aplicaciones. Las fuentes seleccionadas: IEEE Xplore, Science Direct, en el área de Computer Science y ACM Digital Library. Se encontraron pocos trabajos que estén estrechamente relacionados al objetivo de esta RSL, esto puede ser debido a que los procesos de negocio vinculados con el diseño de las aplicaciones y a la usabilidad es un tema poco desarrollado hasta el momento.

El conocimiento extraído de esta RSL será la base para la construcción de un marco de trabajo para el desarrollo de software basado en BPM para la mejora de la usabilidad de aplicaciones [16].

2. Identificación y modelado de los procesos de negocio relevantes del área caso de estudio.

Se seleccionó la plataforma BonitaSoft, que permite la creación de aplicaciones de negocio, mediante la definición del diagrama de procesos de negocio siguiendo la notación BPMN 2.0.

Se seleccionó como caso de estudio el proceso de gestión de viáticos de empleados de un organismo estatal provincial.

Utilizando BonitaSoft, que dispone de tecnologías para la automatización de los procesos de negocio, se continuará

con la digitalización del modelo del proceso en esta plataforma.

Además, se espera elaborar un marco de trabajo basado en el modelado de los procesos de negocio identificados y en técnicas de usabilidad.

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

1. En [17] se resumieron e identificaron los principales problemas y desafíos que afectan a la calidad del software en las disciplinas de entrega continua y pruebas continuas teniendo en cuenta lo obtenido en revisiones sistemáticas de la literatura publicadas en años anteriores.
2. En [18] también se resumieron los factores claves para el éxito al realizar pruebas tipo flaky en servicios REST. Este tipo de pruebas tiene el problema de presentar diferentes resultados en cada ejecución. Por ello es necesario tener en cuenta un conjunto de buenas prácticas y estrategias, con el fin de disminuir sus consecuencias adversas.
3. En [19] se realizó una revisión sistemática de la literatura obteniendo las soluciones ante los problemas en pruebas continuas. Esta disciplina al ser nueva y estar enmarcada en las técnicas de entrega y despliegue continuo necesita ser adecuadamente analizada y revisada.
4. En [20] también se realizó una revisión sistemática de la literatura orientada al cross-browser test. Estos tipos de prueba son actualmente muy costosas, debido principalmente a la gran cantidad de configuraciones y dispositivos. Esto trae como consecuencia que un mismo sitio web sea visto de diferente manera en distintos navegadores. Muchas veces modificando su funcionalidad. Y es uno de los principales desafíos para las

pruebas continuas, debido a este costo en esfuerzo a la vez que debe ser automatizado para lograr resultados beneficiosos.

En general, para esta línea de trabajo se espera desarrollar un modelo de buenas prácticas para la gestión de pruebas continuas en equipos de trabajo que tengan implantados la entrega continua de software. Este modelo hará especial hincapié en los tipos de prueba estudiados y detallados en esta sección.

4. FORMACIÓN DE RECURSOS HUMANOS

En el Grupo de Investigación sobre Calidad de Software (GICS) están involucrados 6 docentes investigadores, 1 tesista de doctorado, 1 becario de investigación de pregrado y 2 tesistas de maestría que finalizaron el cursado en el 2017.

5. REFERENCIAS

- [1] 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.
- [2] Randell, Brian. "Fifty Years of Software Engineering-or-The View from Garmisch." *arXiv preprint arXiv:1805.02742* (2018).
- [3] Kong, Pingfan, et al. "Automated testing of android apps: A systematic literature review." *IEEE Transactions on Reliability* 99 (2018): 1-22.
- [4] R. Pressman, *Ingeniería de Software. Un Enfoque Práctico*, Sexta ed., McGraw Hill, 2005.
- [5] J. A. McCall, P. K. Richards y G. F. Walters, *Factors in Software Quality*, 1997.
- [6] International Organisation for Standardisation, *ISO/IEC 25010, Systems and software engineering -- Systems and software Quality Requirements and Evaluation*

- (SQuaRE) -- System and software quality models; Geneva, Switzerland, 2011.
- [7] Arvanitou, Elvira Maria, et al. "A mapping study on design-time quality attributes and metrics." *Journal of Systems and Software* 127 (2017): 52-77.
- [8] R. Malhotra y A. Jain, «Software Fault prediction for Object Oriented Systems: A Literature Review,» *ACM SIGSOFT Software Engineering Notes*, vol. 36, nº 5, pp. 1-6 , 2011.
- [9] Y. C. Cavalcanti, P. A. da Mota Silveira Neto y . do Carmo Machado, «Towards Understanding Software Change Request Assignment: a survey with practitioners,» de *Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering*, Porto de Galinhas, Brazil, 2013
- [10] L. Chen, "Continuous Delivery: Huge Benefits, but Challenges Too" in *IEEE Software* 03/2015. V. 32(2).
- [11] J. Humble and D. Farley. "Continuous delivery: reliable software releases through build, test, and deployment automation", 1st ed. Boston, US: Pearson Education, 2010.
- [12] O. Prusak. "Continuous Testing: The Missing Link in the Continuous Delivery Process". Blaze Meter. 2015. <https://blazemeter.com/blog/continuous-testing-missing-link-continuous-delivery-process>
- [13] Patricia Bazan. "Implementación de Procesos de Negocio a través de Servicios aplicando Metamodelos, Software Distribuido y Aspectos Sociales". Tesis presentada para obtener el grado de doctor en ciencias informáticas. 2015.
- [14] <https://es.bonitasoft.com/plataforma-bonita>
- [15] Bicevskis, Janis, and Zane Bicevska. "Business process models and information system usability." *Procedia Computer Science* 77 (2015): 72-79.
- [16] Moreno, Juan Carlos, Marcelo Martin Marciszack, and Juan Pablo Fernandez Taurant. "Usability through Business Model." 2016 11th Iberian Conference on Information Systems and Technologies (CISTI). IEEE, 2016.
- [17] Mascheroni, Maximiliano; Irrazábal, Emanuel. *Problemas que afectan a la Calidad de Software en Entrega Continua y Pruebas Continuas*. CACIC 2018
- [18] Mascheroni, M. A., & Irrazábal, E. (2018). Identifying key success factors in stopping flaky tests in automated REST service testing. *Journal of Computer Science and Technology*, 18(02), e16-e16.
- [19] Mascheroni, M. A., & Irrazábal, E. (2018). Continuous Testing and solutions for testing problems in Continuous Delivery: A Systematic Literature Review. *Computación y Sistemas*, 22(3).
- [20] J Sabaren, L. N., Mascheroni, M. A., Greiner, C. L., & Irrazábal, E. (2018). A Systematic Literature Review in Cross-browser Testing. *Journal of Computer Science and Technology*, 18(01), e03.