

Arquitectura de Software en el Proceso de Desarrollo Ágil. Una Perspectiva Basada en Requisitos Significantes para la Arquitectura.

Mg. Mirta E. Navarro¹, Mg. Marcelo P. Moreno², Lic. Juan Aranda³, Lic. Lorena Parra⁴, Lic. José R. Rueda⁵

Departamento de Informática - F.C.E.F. y N. - U.N.S.J.
Complejo Islas Malvinas. Cereceto y Meglioli. 5400. Rivadavia. San Juan
¹mirtaenavarro@yahoo.com.ar ²mpmoren@gmail.com ³juanaranda@live.com ⁴lorenaparra152@yahoo.com.ar ⁵joserocardorueda@hotmail.com

Resumen

Recopilar, comprender y gestionar los requisitos es un aspecto crítico en todos los métodos de desarrollo. Esto también es cierto para las Metodologías Ágiles en la que la captura de requisitos es realizada en todo el proceso de desarrollo, con requisitos que van evolucionando y cambiando a lo largo del ciclo de vida. Este proceso es opuesto al enfoque de la Arquitectura del Software, donde los requerimientos deben ser identificados, recabados y comprendidos en las primeras etapas, ya que cambios posteriores afectan considerablemente el resultado final de la arquitectura. Este tratamiento con perspectivas diferentes, ha sido uno de los factores que ha causado la sensación de que las Metodologías Ágiles y la Arquitectura de Software van en direcciones diferentes y no pueden coexistir.

En este trabajo se presenta una línea de investigación en la que este equipo se ha enfocado en los últimos dos años y tiene como objetivo proponer un modelo que facilite la identificación y captura de los llamados “Requisitos Significantes para la Arquitectura”, permitiendo de esta manera la integración de aspectos arquitectónicos en el proceso de desarrollo de Sistemas de Información con Metodologías Agiles, favoreciendo

de este modo, atributos de calidad y flexibilidad ante los cambios.

Palabras claves: *Software Architecture, Agile methodologies, Architecturally Significant Requirement, Information Systems*

Contexto

El presente trabajo se encuadra dentro del área de I/D de la IS y de los SI, y describe los lineamientos generales del proyecto de investigación: “*Arquitecturas de Software en el Proceso de Desarrollo Ágil, Una Perspectiva Basada en Requisitos*”, iniciado en enero de 2018, con una duración de dos años y que tiene como unidades ejecutoras al Departamento de Informática, FCEFyN de la UNSJ.

El grupo de investigación tiene una trayectoria de 16 años en diferentes proyectos vinculados a Metodologías de Desarrollo y Tecnologías, con numerosas publicaciones en diferentes ámbitos, y con la formación de recursos humanos en el área de interés.

Introducción

Las Metodologías Ágiles (MA) se centran en el trabajo en equipo, la adaptabilidad y colaboración dentro del equipo de desarrollo del software y

también entre los miembros del equipo y los usuarios finales. El uso de las MA, ha marcado una tendencia para su adopción al desarrollo de proyectos de software con necesidades cambiantes y a la espera de beneficios en el menor tiempo posible [1,2,3].

Desde la perspectiva de los requisitos con implicancias en aspectos arquitectónicos en particular, si bien las MA explican que éstos deben estar entre los de mayor prioridad y deben ser tenidos en cuenta desde el comienzo [4,5], pensamos que más que una falta de atención a la definición de los requisitos arquitectónicos al comienzo, la consideración de los mismos es un proceso no exento de dificultades y esto hace que típicamente se evite un trabajo inicial sustancial, en la suposición de que los requisitos siempre cambian, evolucionan y continúan cambiando a lo largo del ciclo de vida del proyecto. En ese sentido, generalmente se identifican y captan los requisitos, tanto funcionales como no funcionales, y luego se intenta definir una arquitectura de software que los cumpla.

La Arquitectura de Software (AS), en tanto, describe la solución de un sistema y por lo tanto tradicionalmente se piensa como una parte temprana de la fase de diseño [6] ya que reúne todos los requerimientos técnicos y operacionales y que son difíciles de cambiar durante el proceso de desarrollo. Es principalmente importante para satisfacer los requisitos no funcionales, que están relacionados a los atributos de calidad como el rendimiento, seguridad y escalabilidad.

Estas decisiones tempranas llevan un peso importante con respecto al desarrollo del resto de un sistema, condicionando decisiones posteriores que implican a su vez, otras ramificaciones subsiguientes. Esto supone una captura de requisitos sin cambios sustanciales en las etapas

intermedias y finales del desarrollo de un proyecto, visión que se contraponen a las dificultades para la identificación de requisitos arquitectónicos que se percibe en las MA.

Este tratamiento con enfoques diferentes en las etapas tempranas (y también en otros aspectos), ha sido uno de los factores que ha causado la sensación de que las MA y la AS van en direcciones diferentes y no pueden coexistir juntas [7]. Sin embargo, en los últimos años esta tendencia está cambiando [8,9], hasta el punto que varios autores hacen referencia al término “Arquitectura Ágil” (AA) [8,9]. En la AA, se enfatiza fuertemente en el concepto de los “Requisitos Significantes para la Arquitectura” (RSA). En tal sentido, aquí se propone la integración de las AS en el ciclo de vida de las MA, desde la perspectiva de la identificación de los RSA considerando diferentes propuestas existentes en el mismo sentido [9,10,11], que son la base para discutir y llevar a cabo más investigaciones sobre RSA. Pensamos que esos métodos y enfoques pueden ser complementarios entre sí, tomando lo mejor de cada uno de ellos, y a la vez podrían ser combinados con los hallazgos que han sido investigados por los autores de este artículo [12] en el área de las metodologías ágiles.

Requisitos de Importancia Arquitectónica.

Un Requisito Significante para la Arquitectura es un requisito que tendrá un efecto importante en la arquitectura, y que, si llegase a estar ausente, la arquitectura resultante será totalmente diferente. Tomando la definición de [10] los RSA “*son aquellos requisitos que tienen un impacto medible en una arquitectura de sistemas de software*”,

por lo que no sería posible diseñar una arquitectura adecuada, si no se identifican y capturan adecuadamente.

Un problema recurrente que se presenta con los RSA [8,9] es que generalmente toman la forma de requisitos no funcionales, pero en algunos casos también la forma de requisitos funcionales. Además, los RSA suelen ser subjetivos, relativos y también interactúan entre ellos. Son subjetivos, porque pueden ser vistos, interpretados y analizados de manera diferente por diferentes personas y en diferentes contextos; son relativos, porque la importancia de cada RSA se determina a veces a partir de su relación con otros RSA en un contexto dado, y se considera que interactúan entre ellos en el sentido de que al intentar alcanzar un RSA particular, puede a su vez afectar (en forma favorable o desfavorable) a otros RSA. Los RSA son más difíciles de comprender que los requisitos funcionales [10], por lo que en general pasan desapercibidos o no obtienen suficiente atención por adelantado, factor que es más acentuado cuando se utilizan MA, donde generalmente se expresan (muchas veces en forma contradictoria) de manera informal durante el análisis de requisitos. También son difíciles de validar cuando el proyecto está finalizado. Por todo lo expuesto, se considera que la identificación de los RSA, es una tarea difícil, que lleva mucho trabajo y es poco clara para los que no son expertos en arquitecturas [9]. Para abordar esta situación, algunos autores, han propuesto una serie de estrategias de identificación y captura de RSA a partir de documentos de requisitos tradicionales de una MA.

Enfoques.

Hay diferentes métodos para identificar y capturar los RSA. Algunos están basados en el conocimiento de los

objetivos del negocio [9], otros en un framework [10] y otros se basan en el conocimiento detallado del dominio [11]. El resumen de esos enfoques se detalla a continuación:

- El enfoque basado en un Framework propone una forma de identificar y capturar RSA a partir de una caracterización previa de los mismos sobre la base de un estudio empírico realizado con expertos.
- El enfoque basado en el Conocimiento del Dominio (CdD) en cambio, hace énfasis en las propiedades del medio ambiente y las suposiciones sobre él, más que en la caracterización de los RSA.
- Finalmente, un enfoque que tiene como propósito centrarse principalmente en los Objetivos de Negocio (ON) y posteriormente a partir de ellos, determinar cómo esos objetivos influyen en los requisitos de calidad que finalmente serán origen de muchos RSA.

Estos enfoques, más algunos otros que pudieran surgir en el periodo de investigación serán tenidos en cuenta en el proyecto y constituyen la base de la propuesta de investigación que se presenta en este trabajo.

Tópico de investigación, Desarrollo e Innovación

Los posibles beneficios de una integración de Arquitectura de Software en las Metodologías Ágiles, desde la perspectiva de la identificación y captura de los RSA, no es un tema lo suficientemente explorado en ambientes académicos, habiéndose encontrado escasas publicaciones relacionadas a los enfoques ya mencionados.

Actividades y Objetivos

Las actividades en las que se ha planificado el desarrollo de la investigación son las siguientes:

- Estudio de factores relevantes en la identificación y captura de los RSA.
- Analizar propuestas alternativas para la identificación y captura de los RSA.
- Fijación de criterios para la especificación de los RSA.
- Adopción de la MA para la integración con los RSA.
- Describir las características que se deben observar en el planteamiento y clasificación de requerimientos funcionales, no funcionales y RSA.

Todas estas actividades serán realizadas con el objetivo de proponer un Modelo de Identificación y Captura de los RSA, que pueda ser incorporado a los procesos de metodologías ágiles, respetando los requerimientos propios del dominio agilista. A partir de ese punto, se continuará con las siguientes actividades:

- Implementación de Sistemas de Información usando el Modelo de identificación/captura de RSA.
- Pruebas, validaciones y elaboración de conclusiones.
- Realización de transferencias al medio, divulgación científica y publicaciones

Formación de Recursos Humanos

El equipo de trabajo está compuesto por cinco docentes-investigadores, y cuatro alumnos adscriptos.

Con los resultados de la presente investigación, se harán actividades de

divulgación en publicaciones y presentaciones en eventos nacionales e internacionales y también asesorar trabajos de tesis de estudiantes de grado y/o posgrado.

En el periodo 2016-2017 se presentaron nueve trabajos en diversas publicaciones en Congresos Nacionales y Revistas Especializadas sobre la temática abordada. Además, se han asesorado los siguientes trabajos vinculados a la temática de referencia:

- Una tesis de maestría finalizada y aprobada en defensa pública (diciembre de 2017).
- Una tesis de grado finalizada y aprobada. (abril 2017).

Referencias

[1] Schwaber, K., Sutherland, J., "The scrum guide. The Definitive Guide to Scrum".En:
<https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>.
Accedido el 20/02/2017.

[2] Cohen, D., Lindvall, M., Costa, P., "An Introduction to Agile Methods" Pdf (2004) En:
http://www.cse.chalmers.se/~feldt/course/s/agile/cohen_2004_intro_to_agile_methods.pdf.
Accedido el 20/02/2018

[3] Breivold, H.P., Sundmark, D., Wallin, P. and Larsson, S., "What Does Research Say about Agile and Architecture?" en Proceedings of the Fifth International Conference on Software Engineering Advances (ICSEA), USA, (2010).

[4] Schwaber, K., Beedle M. "Agile Software Development with Scrum" 1st Edition. Prentice Hall PTR Upper Saddle River, NJ, USA 2001.

- [5] Kenneth Rubin “Essential Scrum: A Practical Guide to the Most Popular Agile Process”. 1st Edition. Addison-Wesley Signature Series. 2013.
- [6] Fowler, M. “Patterns of Enterprise Application Architecture” 1st Edition. Addison-Wesley. 2003.
- [7] Nord RL, Tomayko JE. “Software architecture-centric methods and agile development.” IEEE Software 2006; 23(2):47–53.
- [8] Babar MA, Brown AW, Mistrik I, “Agile Software Architecture Aligning Agile Processes and Software Architectures”. Morgan Kaufmann Publishers Inc. 2013.
- [9] Bass L, Clements P, Kazman R. “Software architecture in practice” 3rd ed. Addison-Wesley. 2012
- [10] Chen L, Babar MA, Nuseibeh B. “Characterizing architecturally significant requirements.” IEEE Software 2013; 30:38–45.
- [11] Azadeh Alebrahim, Maritta Heise. “Intertwining Relationship Between Requirements, Architecture, and Domain Knowledge”. ICSEA 2014: The Ninth International Conference on Software Engineering Advances.
- [12] Navarro M., Moreno M., Aranda J., et al. “Integración de Arquitectura de Software en el Ciclo de Vida de las Metodologías Ágiles.” Una Perspectiva Basada en Requisitos. XIX Workshop de Investigadores en Ciencias de la Computación (WICC 2017). (566-569). ISBN 978-987-42-5143-5.