

Propuesta metodológica para la gestión de proyecto de software ágil basado en la *Web*

Pedro Luis Alfonzo¹, Sonia Mariño^{1,2} y María Viviana Godoy¹

¹Facultad de Ciencias Exactas y Naturales y Agrimensura, ²Facultad de Humanidades, Universidad Nacional del Nordeste. Argentina

plalfonzo@hotmail.com; simarinio@yahoo.com; mvgodoy@exa.unne.edu.ar

Resumen

La gestión de proyectos es una parte esencial de la Ingeniería del Software. Aun cuando no garantiza su éxito, usualmente una mala gestión conlleva al fracaso. Además, la selección adecuada de una metodología es trascendental para el éxito de un proyecto. En este trabajo se presenta una propuesta metodológica integradora que permite el desarrollo de software accesible desde la web. Se partió de las prácticas expuestas en dos metodologías validadas empíricamente. Una de ellas diseñada ad-hoc y utilizada en la construcción de aplicaciones *Web* para diversos sectores: culturales, educativos, administrativos, productivos, entre otros. La segunda es SCRUM, una metodología ágil para gestionar el proceso de desarrollo del software. Se estima lograr mejores resultados, considerando que SCRUM se emplea de manera efectiva en diversos proyectos y la metodología desarrollada ad-hoc generó productos transferibles y en uso.

Palabras clave: ingeniería del software, desarrollo ágil, generación de software, gestión de procesos.

Methodological Proposal for Managing an Agile Software Web-based Project

Abstract

Project management is an essential part of software engineering. Although it does not guarantee its success, poor management usually leads to failure. In addition, proper selection of a methodology is crucial for a project's success. This paper presents an integrative methodological approach that permits developing web-accessible software. The proposal is based on practices defined in two empirically validated methodologies. One was ad-hoc designed and used to build web applications applied in various sectors. The second is SCRUM, an agile methodology for managing the software development process. This is thought to achieve better results, considering that SCRUM is used effectively in diverse projects and the methodology developed ad-hoc generated transferable in use products.

Keywords: software engineering, methodology, agile development, software generation, process management.

1. Introducción

Las transformaciones tecnológicas y la evolución de las organizaciones incidieron en el diseño y desarrollo de proyectos informáticos. La informática ha recorrido un cambio paradigmático, progresando desde sistemas centralizados basados en mainframe hasta las computadoras personales conectadas a Internet.

En tal sentido, se considera que una decisión importante a la hora de abordar un proyecto informático es la metodología de desarrollo Software que se utilizará. Cabe aclarar que no siempre se ha de aplicar la misma para todo tipo de proyectos, sino que es conveniente analizar las necesidades para determinar cuál es la más apropiada.

En este mismo orden de ideas Canos *et al.* (2003) plantean que las metodologías tradicionales llevan asociadas un marcado énfasis en el control del proceso mediante una rigurosa definición de roles, actividades y artefactos, incluyendo modelado y documentación detallada, y éstas han demostrado ser efectivas y necesarias en proyectos de gran tamaño, por lo que, este enfoque no resulta ser el más adecuado para muchos de los proyectos actuales, donde el entorno del sistema es muy cambiante y el tiempo de desarrollo se reduce.

En consecuencia las metodologías ágiles emergen como una posible respuesta ante el referido escenario, por

estar especialmente orientadas a proyectos pequeños. Estos constituyen una solución a la medida para esos entornos, ya que aportando una elevada simplificación, pero sin renunciar a las prácticas esenciales para asegurar la calidad del producto (Canós *et al.*, 2003). Se mencionan las comprendidas en esta categoría: i) Agile Database Techniques (AD), ii) Agile Modeling (AM), iii) Adaptive Software Development (ASD), iv) Agile Unified Process (AUP), v) Crystal, vi) Feature Driven Development (FDD) vii) Dynamic Systems Development Method (DSDM) viii) Lean Software Development, ix) SCRUM, x) Test-Driven Design (TDD) y xi) eXtreme Programming (XP).

Considerando que la selección adecuada de una metodología es trascendental para el éxito de un proyecto en Mariño y Godoy (2008a) se presenta una diseñada para la generación de entornos virtuales, como apoyo complementario al proceso de enseñanza-aprendizaje en la Educación.

Reafirmando con esto y aunado por lo planteado por Sommerville (2005) se afirma que la gestión de proyectos de software es una parte esencial de la ingeniería del software (IS). Ésta no garantiza el éxito del proyecto, sin embargo usualmente una mala gestión lleva su fracaso. El software es entregado tarde, los costos son mayores que los estimados y los requerimientos no se cumplen. Por lo

tanto, la gestión efectiva de un proyecto de software depende de planificar completamente su progreso, debido a que la IS siempre está sujeta a restricciones organizacionales de tiempo y presupuesto.

En este sentido, la propuesta metodológica presentada en este trabajo consiste en complementar las prácticas de la metodología diseñada ad-hoc expuestas en Mariño y Godoy (2008a) y Mariño y Godoy (2008b); y utilizada en la construcción de aplicaciones Web como los trabajos mencionados en Alderete *et al.* (2010), Mariño *et al.* (2009) y Mariño *et al.* (2010), con la metodología ágil SCRUM, para la gestión y control del proceso de desarrollo de software, a fin de lograr la entrega rápida de software de alta calidad.

El artículo está organizado como sigue. En la sección 2 se describe la metodología utilizada en la elaboración de este trabajo. La sección 3 sintetiza la metodología diseñada ad-hoc y aplicada en la construcción de aplicaciones Web, para entornos de enseñanza-aprendizaje. La sección 4 se centra en SCRUM, presentando sus características, el proceso que sigue y las prácticas propuestas. En la sección siguiente se describe la propuesta metodológica que permite a priori la gestión y control del proceso de desarrollo de software detallándose sus principales características. Finalmente se exponen las conclusiones y futuras líneas de trabajo.

2. Metodología

La metodología aplicada en este trabajo es de tipo exploratorio. Se basó en las siguientes etapas:

- Relevamiento, selección y estudio de metodologías ágiles para el desarrollo de software.
- Revisión de antecedentes de la utilización de SCRUM en la gestión y control de proyectos.
- Elaboración de una propuesta integradora que permita desarrollar de manera exitosa aplicaciones Web. Lo expuesto en este trabajo combina la metodología diseñada ad-hoc, basada en el desarrollo de prototipos incrementales o evolutivos y validado en numerosas experiencias de desarrollo de software accesible desde la web sintetizada en Mariño y Godoy, (2008a), con la metodología ágil SCRUM, que proporciona un marco regulatorio para la gestión de proyecto. Además, esta última administra el entorno de trabajo, atiende los recursos humanos de la organización productora y los requerimientos de los Stakeholders.

3. Metodología diseñada ad-hoc

En Mariño y Godoy (2003a), Mariño y Godoy (2008a) se presenta la metodología diseñada ad-hoc, a partir de la revisión de metodologías clásicas para el desarrollo de software tradicional (Mariño y Godoy, 2003b y Pressman, 2005) y de software basado en arquitectura web (Ochoa *et al.* 2008). Se evaluaron y adoptaron aquellas más apropiadas para su implantación en sectores culturales, educativos, administrativos, productivos, entre otros y sintetizada en una propuesta de Mariño y Godoy (2008b). Ésta se fundamentó en el desarrollo de prototipos incrementales o evolutivos referenciados por Corcos (2000) y Pressman (2005), aplicando etapas particulares a fin de responder a requerimientos de la ingeniería web y de los demandantes.

En la Tabla 1 se presentan las fases, actividades y tareas contempladas. Una descripción detallada puede observarse en Mariño y Godoy (2008b).

4. Gestión de proyectos ágil con SCRUM

SCRUM es una metodología para la gestión y control de proyectos, centrada en la construcción de software que satisface las necesidades del cliente, cumple con los objetivos del negocio y el equipo de desarrollo que construye el producto. Al no establecer prácticas de IS, se combina fácilmente con otras metodologías de desarrollo.

Díaz (2009) define a SCRUM, como una colección de procesos para la gestión de proyectos, que permite centrarse en la entrega de valor para el cliente y la potenciación del equipo para lograr su máxima eficiencia, dentro de un esquema de mejora continua.

Al considerar lo planteado por Sutherland *et al.*, (2007) se puede afirmar que la gestión de proyectos ágil con SCRUM derivó de mejores prácticas de negocios en empresas como Toyota, Fuji-Xerox, Honda y Canon. Toyota consigue habitualmente cuatro veces la productividad y 12 veces la calidad de los competidores.

En Schwaber (1995) se mencionan algunas implementaciones de SCRUM, como ser las diversas variantes de éste enfoque para el desarrollo de nuevos productos, que fue observada por primera vez por Takeuchi y Nonaka (1986) en el Fuji-Xerox, Canon, Honda, NEC, Epson, Brother, 3M, Xerox y Hewlett-Packard. Un enfoque similar aplicado al desarrollo de software en Borland, se indicó en Coplien (1994).

Un enfoque de este proceso lo aplicó Sutherland (1996) al desarrollo en Smalltalk y Schwaber (1996) a la producción en Delphi. SCRUM es utilizado por empresas

Tabla 1. Fases de la metodología propuesta.

| Fases | Actividades | Tareas |
|---------------|--|--|
| Planificación | Planificación del proyecto. | <ul style="list-style-type: none"> • <i>Análisis del entorno.</i> • <i>Análisis de factibilidad.</i> • <i>Selección de herramientas ad-hoc</i> • <u>Identificación de roles.</u> • <u>Creación de la Pila de productos priorizada (Product Backlog).</u> • <u>Definición del alcance.</u> • <u>Estimación del esfuerzo.</u> |
| | Planificación de la iteración/versión. | <ul style="list-style-type: none"> • <u>Selección de requerimientos a incluir en el Sprint (Sprint Backlog)</u> • <u>Definición de las tareas para cada requerimiento.</u> • <u>Estimación del esfuerzo para cada tarea.</u> |
| Desarrollo | Análisis. | <ul style="list-style-type: none"> • <i>Recopilación de información.</i> • <i>Selección y preparación de contenidos.</i> |
| | Diseño e Implementación. | <ul style="list-style-type: none"> • <i>Diseño del entorno. (Prototipo de interfaces).</i> • <i>Validación del prototipo.</i> • <i>Implementación del prototipo.</i> |
| | Validación y prueba. | <ul style="list-style-type: none"> • <i>Validación y prueba del prototipo construido.</i> |
| Integración | Integración y pruebas. | <ul style="list-style-type: none"> • <i>Integración de componentes y pruebas.</i> • <i>Entrega de la versión.</i> |
| Entrega | Entrega Final. | <ul style="list-style-type: none"> • <i>Prueba del sistema.</i> • <i>Entrega de la versión final.</i> |

grandes y pequeñas, incluyendo Yahoo, Microsoft, Google, Lockheed Martin, Motorola, SAP, Cisco, GE, CapitalOne y la Reserva Federal de los EE.UU (Deemer *et al.*, 2010).

Como método ágil:

- Es un modo de desarrollo adaptable, antes que predictivo.
- Está orientado a las personas, más que a los procesos.
- Emplea el modelo de construcción incremental basado en iteraciones y revisiones.
- En Canós *et al.* (2003) se resumen sus principales características:
- Especialmente indicada para proyectos con un rápido cambio de requisitos.
- El desarrollo de software se realiza mediante iteraciones, denominadas *Sprint*, con una duración de 30 días. El resultado de cada uno de ellos es un incremento ejecutable que se muestra al cliente.
- La realización de reuniones a lo largo del proyecto, entre ellas se destaca la diaria del equipo de desarrollo, con una duración aproximada de 15 minutos y con miras a la coordinación e integración de actividades.

Las prácticas empleadas por SCRUM para mantener un control ágil en el proyecto son: i) Revisión de las itera-

ciones, ii) Desarrollo incremental, iii) Desarrollo evolutivo, iv) Auto-organización del equipo y v) Colaboración.

Los roles, artefactos y eventos principales se resumen en la Figura 1.

Discusión y resultado

El desarrollo de aplicaciones *Web* se diferencia de la generación de aplicaciones o software tradicional en características tales como las mencionadas por Rossi *et al.*, (2007): i) son evolutivas, tanto en sus requerimientos como en su funcionalidad; ii) los tiempos de desarrollo suelen ser más cortos; iii) el proceso de desarrollo de las aplicaciones *Web* es incremental, no “termina” como en los proyectos tradicionales de software, entre otros.

Teniendo en cuenta éstas características y complementando por Fitsilis, (2008) se afirma que la gestión de proyectos ágil de software se basa en los siguientes principios: abrazar el cambio, centrarse en el valor para el cliente, entregar parte de la funcionalidad de forma incremental, colaborar, reflexionar y aprender continuamente, se optó por aplicar los principios de la metodología ágil SCRUM para la gestión y control del proceso de desarrollo de la metodología presentada en la sección 3, y de esta manera desarrollar la propuesta metodológica que permita el desarrollo ágil de software basado en *Web*.

Las actividades de gestión a incluir en la propuesta son: i) planificación; ii) estimación de tiempos y costos; iii) la gestión de personal y iv) la gestión de riesgos. Para lo cual se introdujeron las prácticas y artefactos SCRUM que se mencionan:

- La gestión de los requerimientos del sistema (*Product Backlog*).
- Priorización, estimación y definición del alcance de cada funcionalidad requerida.
- La gestión del *Sprint* (*Sprint Backlog*) y estimación del esfuerzo de cada tarea, incluido el diseño del prototipo.
- Las reuniones y roles.
- Gráficos de *burn-down*: para gestionar y seguir el avance de cada *Sprint*.

Para especificar los requerimientos de software por parte del cliente se utilizan los casos de uso, práctica utilizada en la metodología desarrollada *ad-hoc* (Mariño y Godoy, 2008a).

El *Sprint*: está compuesto por la fase de desarrollo, donde el software se construye en base a prototipos, previamente diseñados y validados de acuerdo a los requerimientos incluidos en el *Sprint Backlog*. La duración será decidida por el equipo.

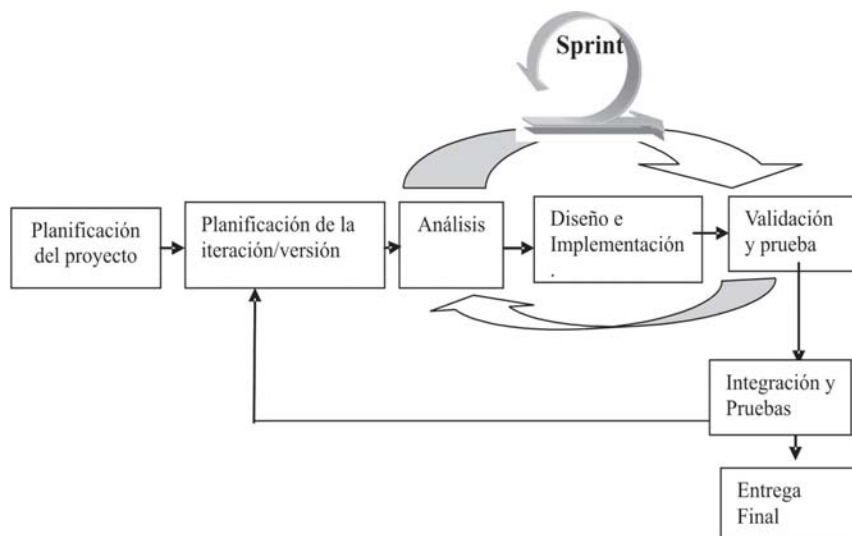
El Gráfico de *Burn-Down*, se prevé su utilización para visualizar el avance de las tareas y del *Sprint*.

La Figura 2 ilustra el marco de trabajo metodológico propuesto, que integra las prácticas y artefactos de SCRUM, mencionadas anteriormente, con las prácticas



Fuente: Deemer *et al.*, (2009).

Figura 1. Roles, artefactos y eventos principales de SCRUM.



Fuente: elaboración propia.

Figura 2. Marco de trabajo ágil para gestionar el proceso de desarrollo del software.

de desarrollo de software incorporadas en la metodología diseñada *ad-hoc* (Mariño y Godoy, 2008a).

Como se visualiza, el proceso de iteración del *Sprint* se representa en las tareas que componen la fase de desarrollo (Tabla 1), es decir la generación de una versión del prototipo.

En la Tabla 1, se sintetizan las actividades y tareas a llevar a cabo en cada una de las fases propuestas en la construcción de un producto software. Las tareas propias de SCRUM que fueron agregadas, son las mencionadas anteriormente en esta sección y se encuentran subrayadas; y las que componen la metodología diseñada *ad-hoc* se señalan en letras cursivas.

6. Consideraciones finales y trabajos futuros

Actualmente, se relevan y estudian metodologías ágiles para gestionar y desarrollar proyectos de software basados en plataforma *Web*. En este trabajo, se expuso una propuesta generada como superación de una diseñada *ad-hoc* probada previamente, para ser tratada a través de las prácticas y actividades de una metodología ágil como SCRUM.

Se presentó un marco de trabajo metodológico, se describieron las actividades y tareas involucradas en cada fase, complementando ambas metodologías preexistentes a fin de gestionar y controlar el proceso de desarrollo del software. Además de agilizar la entrega de productos software de calidad y adaptable a los requerimientos del cliente.

Se prevé su aplicación a casos de estudio o experiencias para la implementación de la metodología propuesta, en el contexto de influencia de la Universidad Nacional del Nordeste, a fin de validarla y aportar información de realimentación.

Referencias

- ALDERETE, Romina; ESCALANTE, Jaquelina; MARIÑO, Sonia; GODOY, María (2010). Software educativo: apoyo para Lengua en el nivel primario. **Encuentro Internacional BTM 2009**. Uruguay.
- CANÓS, José; LETELIER, PatricioL; PENADÉS, María. (2003). Metodologías Ágiles en el Desarrollo de Software. Universidad Politécnica de Valencia. Disponible en: <http://www.willydev.net/descargas/prev/TodoAgil.pdf>. Consulta: 20 de agosto del 2010.
- COPLIEN, James (1994). Borland Software Craftmanship: A New Look at Process, Quality and Productivity. Proceedings of the **5th Annual Borland International Conference**, June 5, 1994. Orlando, Florida.
- CORCOS, Daniel (2000). El Modelo Espiral. **Cuaderno de Reportes Técnicos en Ingeniería del Software Nro 3**. (Recatalogado como RTIS Volumen 2, Nro 1, Año 2000). 29-40 pp.
- DEEMER, Pete; BENEFIELD, Gabrielle; LARMAN, Craig; VODDE, Bas (2010). The Scrum Primer. Versión 1.2. Scrum Training Institute. Disponible en: <http://goodagile.com/scrumprimer/scrumprimer.pdf>. Consulta: 20 de noviembre del 2011.
- DEEMER, Pete; BENEFIELD, Gabrielle; LARMAN, Craig; VODDE, Bas (2009). Información Básica de Scrum the Scrum Primer Version 1.1. Scrum Training Institute. Traducción de Leo Antoli. Agile-Spain. Disponible en: http://www.goodagile.com/scrumprimer/scrumprimer_es.pdf. Consulta: 30 de mayo del 2011.
- DÍAZ, Ramón (2009). Las metodologías ágiles como garantía de calidad del software. **Revista Española de Innovación, Calidad e Ingeniería del software**, Vol.5, Nº 3, 2009.
- FITSILIS, P. (2008). Comparing PMBOK and Agile Project Management software development processes, in **Advances in Computer and Information Sciences and Engineering**. Sobh,T. (editor). Netherlands: Springer, pp. 378-383.
- MARIÑO, Sonia; GODOY, María V. (2003a). Sistemas de gestión de información universitaria basados en arquitectura web. Congreso Argentino de Ciencias de la Computación. CACIC 2003. La Plata. Argentina.
- MARIÑO, Sonia; GODOY, María V. (2003b). ¿Un aporte hacia la gestión del conocimiento?. Desarrollo de sistemas de información universitarios en la Web. CICIC 2003. Congreso Internacional de la Sociedad de la Información y el Conocimiento. Universidad Pontificia de Salamanca (Campus Madrid).
- MARIÑO, Sonia; GODOY, María V.; BUSSO, Lorena (2009). Selección y evaluación de herramientas FLOSS para la construcción de una plataforma orientada a la centralización y gestión de información educativa. **Revista de Informática Educativa y Medios Audiovisuales**. ISSN 1667-8338. Editorial o entidad responsable: Laboratorio de Informática Educativa y Medios Audiovisuales Facultad de Ingeniería. 6(13): 21-28. Univ. Bs. As.
- MARIÑO, Sonia; GODOY, María V. (2008a). Desarrollo de entornos virtuales educativos. Contribuciones desde el Área de Ingeniería Web. **Quaderns digitals**. Revista electrónica Número 53. ISSN: 1575-9393. Junio 2008.
- MARIÑO, Sonia; GODOY, María V. (2008b). Tecnologías de la información y comunicación como herramientas del desarrollo local. Proyecto acreditado por la Secretaría General de Ciencia y Técnica. Universidad Nacional del Nordeste.
- MARIÑO, Sonia; GODOY, María V., LEZCANO, Javier; ZACARÍAS, Gustavo (2010). Innovaciones en expresiones culturales desde la Universidad. Las TIC como aporte al desarrollo musical de la región NEA. **II Encuentro Sudamericano sobre Gestión Cultural y Participación Ciudadana**. San Luis. Argentina.
- OCHOA, María; BRITOS, Paola; FERNÁNDEZ, Enrique; GARCÍA MARTÍNEZ, Ramón (2008). **Metodologías de Ingeniería Informática**. Ed. Nueva Librería. 869 p.
- PRESSMAN, Roger (2005). **Ingeniería del Software un enfoque práctico**. Ed. Mc Graw-Hill Interamericana. Edición Sexta. 980 p.

- ROSSI, Gustavo; PASTOR, Oscar; SCHWABE, Daniel; OLSINA, Luis (2007). Web Engineering: Modelling and Implementing Web Applications (Human-Computer Interaction Series). Parte III, cap., 5-6 y 13-14.
- SCHWABER, Ken (1996). Controlled Chaos: Living on the Edge. American Programmer, April 1996.
- SCHWABER, Ken (1995). Scrum Development Process, in **OOPSLA Business Object Design and Implementation Workshop**, J. Sutherland, D. Patel, C. Casanave, J. Miller, and G. Hollowell, Eds. London: Springer, 1995.
- SOMMERVILLE, Ian (2005). **Ingeniería del Software**. 7ª Edición. Ed. Pearson.
- SUTHERLAND, Jeff; VIKTOROV, Anton; BLOUNT, Jack; PUNTIKOV, Nikolai (2007). Distributed Scrum: Agile Project Management with Outsourced Development Teams, hicss, pp.274a, **40th Annual Hawaii International Conference on System Sciences (HICSS'07)**, 2007.
- SUTHERLAND, Jeff (1996). ScrumWeb Home Page: A Guide to the SCRUM Development Process Jeff Sutherland's **Object Technology Web Page**.
- TAKEUCHI, Hirotaka; NONAKA, Ikujiro (1986). The New New Product Development Game. **Harvard Business Review**.
-