

Simulando Proyectos de Desarrollo de Software que incorporan Test Driven Development

Diego Alberto Godoy^a, Henry Kotynski^b, Edgardo A. Belloni^c, Eduardo O.Sosa^d,
Mariela Gamonal^e, Nahuel Regueira^f

Centro de Investigación en Tecnologías de la Información y Comunicaciones (C.I.T.I.C.)
Departamento de Ingeniería y Ciencias de la Producción-Universidad Gastón Dachary
Av. López y Planes 6519- Posadas, Misiones, Argentina. Teléfono: +54-376-4438677

^a diegodoy@citic.ugd.edu.ar, ^b henrykotynski@citic.ugd.edu.ar, ^c ebelloni@ugd.edu.ar,
^d eduardo.sosa@citic.edu.ar, ^e mariela.gamonal@citic.edu.ar,
^f nahuel.regueira@citic.edu.ar

Resumen

En este artículo se presenta una línea de investigación denominada Simulación como Herramienta para la Mejora de los Procesos de Software Desarrollados con Metodologías Ágiles Utilizando Dinámica de Sistemas, cuyo objetivo es construir modelos de simulación utilizando la disciplina de dinámica de sistemas, que permitan estudiar los procesos de desarrollo de software ágiles gestionados por diversas metodologías y prácticas ágiles. Particularmente se presentan los avances realizados hasta ahora en relación a construcción de un Modelo de simulación para estudiar el comportamiento de proyectos de software que utilizan Test Driven Development (TDD) en su proceso de desarrollo. Para ello se ha realizado una encuesta en la comunidad de desarrolladores sobre el uso que le dan a TDD en sus proyectos y cuáles son las características de los mismos. Se presenta además el Subsistema de planificación de pruebas de aceptación, uno de los subsistemas más importantes dentro de modelo que se pretende construir.

Palabras claves: Administración de Proyectos de Desarrollo de Software; Test Driven Development Dinámica de Sistemas.

Contexto

El trabajo presentado en este artículo tiene como contexto marco el proyecto de investigación denominado “Simulación como Herramienta para la Mejora de los Procesos de Software Desarrollados con Metodologías Ágiles Utilizando Dinámica de Sistemas”, registrado actualmente en la Secretaría de Investigación y Desarrollo de la Universidad Gastón Dachary (UGD) [†] y radicado en el Centro de Investigación en Tecnologías de la Información y Comunicaciones de dicha universidad.

El mismo fue ratificado e incorporado como proyecto aprobado en el llamado a presentación interna de la UGD de proyectos de investigación N°6 mediante la Resolución Rectoral 24/A/15.

Entre las líneas con mayores resultados dentro del proyecto referido, se encuentran las de: (i) “Modelo de Simulación Dinámico de Gestión de Proyectos de Desarrollo de Software que utilizan Programación Extrema”, (ii) “Modelo de Simulación Dinámico de Gestión de Proyectos de Desarrollo de Software Bajo Scrum” y (iii) “Modelo de Simulación Dinámico de Gestión de Proyectos de Desarrollo de Software desarrollados con Crystal”. Este artículo se enfoca en la presentación de una nueva

[†] Mediante Res.Rectoral UGD N° 04/I/12.-

línea para estudiar la incorporación de prácticas de Test Driven Development a los proyectos de software a través de un modelo dinámico de simulación.

Introducción

El proceso involucrado en el desarrollo de software es complejo e implica grandes retos para el equipo de trabajo. Uno de los desafíos más grandes es garantizar la calidad del producto y mejorar la productividad a lo largo del ciclo de vida del mismo, desde el comienzo del desarrollo hasta las etapas de mantenimiento [1].

Es difícil poder determinar las características de un producto de software al inicio de su producción. El producto final se va conformando a lo largo de su propio ciclo de vida, este va sufriendo modificaciones constantes, adecuándose a las necesidades actuales del cliente.

Estos continuos cambios de requisitos, incluso en etapas tardías del desarrollo, requieren modelado adicional [2]. Causando la incorporación de código nuevo, lo cual tiende potencialmente a introducir errores, ocasionando que funcionalidades ya implantadas comiencen a fallar. Esto provoca un aumento de costo y tiempo, lo cual es uno de los dilemas principales al momento de encarar un proyecto de software.

Actualmente una de las soluciones a estos problemas es la utilización de metodologías ágiles, que incorporen herramientas que agilicen la ejecución de grandes cantidades de casos de pruebas en cuestión de minutos.

Test Driven Development (TDD) es una práctica iterativa de diseño de software orientada a objetos, que fue presentada por Kent Beck y Ward Cunningham como parte de XP [3]. Ha tenido una gran difusión en los últimos años, debido a que su objetivo es obtener código limpio que trabaje.

Esta es una técnica de programación que consiste en guiar el diseño de una aplicación por medio de pruebas unitarias. Cambia el orden tradicionalmente establecido, de manera

que primero se definen las pruebas y a partir de estas se va desarrollando la funcionalidad; repitiendo el ciclo, de acuerdo a lo que se espera que haga el software, por medio de integraciones y refactorizaciones [4].

Al utilizar la técnica TDD en lugar de realizar complejos diseños y desarrollos, se incorpora solamente las funciones que el cliente necesita evitando errores de complejidad. Sin embargo, la utilización de dicha técnica puede traer beneficios en cuanto a costos y tiempo dependiendo del tipo y tamaño de proyecto de software a desarrollar.

Se han realizado una gran cantidad de trabajos evaluando la eficacia de TDD, efectuando un análisis económico de costo-beneficio sobre la utilidad de esta práctica. En algunos de los estudios se ha llegado a la conclusión que TDD suele ser más beneficiosa para ciertos proyectos, de acuerdo a su tipo, tamaño y complejidad.

Al día de hoy no existe una métrica para determinar cuándo es conveniente utilizar esta técnica. Es por ello que la elaboración e implementación de un modelo que permita simular el desarrollo de un proyecto de software bajo la técnica de programación TDD, proporcionaría una herramienta informática fundamental a los desarrolladores de software a la hora de definir si adoptar o no este método de desarrollo.

Actualmente existen diferentes trabajos y herramientas que permiten simular la administración de proyectos de software, entre ellos se puede citar a los siguientes: “Dynamics of Agile Software Development” [1], “Modeling Agile Software Maintenance Process Using Analytical Theory of Project Investment” [2], “Modelo Dinámico de Simulación de Proyectos de Software con XP” [3] y [4], “Agile Project Dynamics: A Strategic Project Management Approach to the Study of Large-Scale Software Development Using System Dynamics” [5].

Estas herramientas presentan modelos basados en diferentes metodologías ágiles, en los cuales son

propuestos y evaluados diferentes escenarios, algunos de ellos genéricos. No obstante, en ninguno de tales trabajos se presenta un estudio que contemple las prácticas y variables particulares de TDD.

Objetivos Propuestos

Teniendo en cuenta lo planteado en la introducción en este trabajo se presentan los primeros avances en el diseño de un Modelo de Simulación Dinámico basado el modelado sistémico que de soporte a la gestión de procesos de desarrollo de software utilizan prácticas de TDD.

El modelo permite, que los administradores de proyectos, puedan evaluar qué impacto tendrán sus decisiones de gestión sobre el mismo, a lo largo del desarrollo. De esta manera el modelo refleja el efecto del uso de prácticas de TDD en proyectos de desarrollo de software. Formalmente los objetivos de la línea presentada son los siguientes:

Como objetivo general se plantea: “Construir un modelo dinámico que permita simular el desarrollo de un proyecto de software bajo la práctica Test-Driven Development para determinar la eficacia de su utilización de acuerdo a las características del proyecto”.

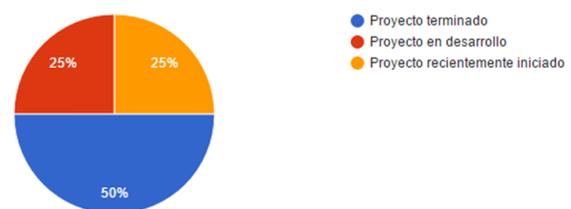
Como objetivos específicos se plantean los siguientes: A) Estudiar el funcionamiento y aplicación del método Test Driven Development. B) Analizar la bibliografía referente a la técnica Dinámica de Sistemas. C) Determinar las variables endógenas y exógenas que intervengan en el proceso de TDD. D) Modelar la estructura del sistema y su comportamiento mediante dinámica de sistemas. E) Verificar y validar el modelo con los datos provenientes de investigaciones y proyectos desarrollados con TDD. F) Evaluar la conveniencia de la adopción de TDD, de acuerdo a los distintos tipos de proyectos, según los resultados obtenidos.

Construcción del Modelo

Primeramente se realizó una encuesta para conocer características de proyectos desarrollados con Test Driven Development. Esta encuesta fue publicada en grupos o comunidades de desarrolladores que trabajan con la metodología, tanto para los de habla hispana como inglesa. Se consultó sobre cuestiones como: duración del proyecto, tamaño del equipo, grado de avance del proyecto, nivel de experiencia en TDD de los desarrolladores, etc.

En la Figura 1 se muestran algunos resultados

Grado de desarrollo del proyecto



Duración del proyecto (promedio)

18 meses

Porcentaje de uso de TDD en el proyecto

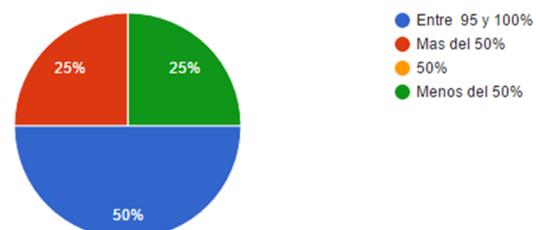


Figura 1: Algunos resultados de la encuesta realizada

En el desarrollo y construcción del modelo se utilizó la herramienta CASE VenSim PLE 5.4c (Versión Académica) [6] seguido las etapas de la Metodología de Dinámica de Sistemas [7], de esta forma, en la Fase de Conceptualización se construya el Diagrama Causal, incluyendo las variables que representan un proyecto de software Scrum y sus interrelaciones. Luego, en la Fase de Formulación se traduce el Diagrama referido a un Diagrama de Forrester, el cual ha sido dividido en Subsistemas Conservativos de

acuerdo con [8]. Finalmente, en la Fase de Evaluación se realizan las corridas de validación y experimentales del modelo.

A continuación se describe uno de los subsistemas principales del modelo, el denominado Subsistema de planificación de pruebas de aceptación. Su diagrama de influencias se puede ver en la Figura 1.

Este subsistema representa el proceso de planificación de cada una de las pruebas de aceptación lo cual es el primer paso para el desarrollo de software a través de esta metodología.

Una prueba de aceptación es la descripción del escenario desde el lado del usuario, que establece como debe funcionar y que debe realizar el sistema, módulo o submódulo, dependiendo de la generalidad de esta.

Las variable exógena que aparecen en este subsistema son Funcionalidades planificadas por iteración y Cantidad de iteraciones, cuanto mayor sea el número de estas variables mayor será el flujo de funcionalidades planificadas. Sin embargo la variable Tiempo promedio de planificación por PA tiene una relación negativa con el Flujo de funcionalidades planificadas. Al aumentar la variable endógena Funcionalidades a desarrollar, y la variable exógena Cantidad promedio de PA a desarrollar por funcionalidad, aumenta el Flujo de planificación por PA y en consecuencia la cantidad de PA a desarrollar, sin embargo este flujo se ve condicionado por el Tiempo promedio de planificación por PA.

La salida de este subsistema es la cantidad de PA a desarrollar.

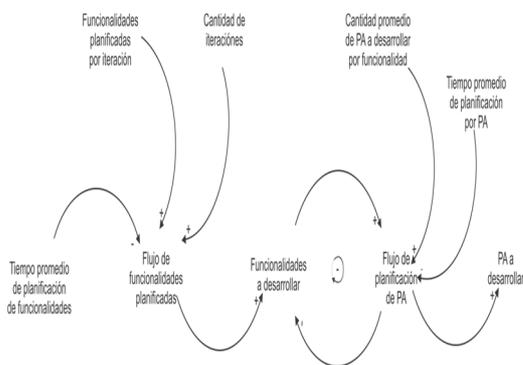


Figura 2: Subsistema de planificación de pruebas de aceptación

Resultados

En este trabajo se presentan avances en la construcción de un “Modelo de Simulación Dinámico para estudiar la incorporación de prácticas de Test Driven Development a los proyectos de software”. Al enfocarse en las particularidades de *TDD*, este trabajo se diferencia de otros en los que se modelan proyectos con Metodologías Tradicionales y en los que se modelan las generalidades de las Metodologías Agiles [1] [2], o bien se modela la especificidad de una de éstas últimas como en [3], [4], [9] [10] [11] [12] y [13].

Por otro lado la realización de una encuesta a desarrolladores permitió entender y conocer el grado de uso de las prácticas de TDD en los proyectos.

Como trabajo futuro se espera avanzar, con la construcción de los demás subsistemas restantes, como por ejemplo: el de desarrollo e implementación de Pruebas de Aceptación, Subsistema de planificación y desarrollo de Pruebas Unitarias, Subsistema de desarrollo de código entre otros. Otra actividad prevista es la de compilar datos de proyectos de desarrollo de software en donde se utilizó TDD, con el fin obtener datos de calidad y cantidad proyectos que siguen estas prácticas. Adicionalmente, se prevé comparar la utilización de los simuladores de modelos para Scrum [10] [11] [9] [12] y en XP [3] y [4], desarrollados anteriormente.

Formación de Recursos Humanos

El equipo de trabajo se encuentra formado por cuatro investigadores con distintos niveles de posgrado, un Doctor en Ciencias Informáticas, un Doctorando en Tecnologías de la Información y Comunicaciones, Magister y Especialista en Ingeniería de Software, un Maestrando de Ingeniería de la Web, Un maestrando en Ingeniería de Software, un investigador con nivel de grado, y seis estudiantes en período de realización de trabajos finales

de grado en el contexto de las carreras de Lic. en Sistemas de Información y de Ingeniería en Informática de la UGD. Actualmente, el número de tesinas de grado aprobadas en el contexto de este proyecto, es de tres, y otras dos en proceso de desarrollo. El número de tesis de maestría terminadas relacionadas con este proyecto es de una.

Bibliografía

- [1] Kim E. Van, Kishore Sengupta, and Luk N. Van., "Dynamics of Agile Software Development," *International Conference of the System Dynamics Society*, 2009.
- [2] Konga, Li, Liu y Chen, Jing Xiaoying, "Modeling Agile Software Maintenance Process Using Analytical Theory of Project Investment," *International Conference on Advances in Engineering 2011*, 2011.
- [3] Tamara Kasiak y Godoy Diego Alberto, "Simulación de Proyectos de Software desarrollados con XP," *XIV Workshop de Investigadores en Ciencias de la Computación.*, 2012.
- [4] Godoy Diego Alberto y Kasiak Tamara., "Modelo dinámico de simulación para la gestión de proyectos de software desarrollados con XP," in *Actas XVIII Congreso Argentino de Ciencias de la Computación*, 2012, p. 10.
- [5] Firas Glaiel, "Agile Project Dynamics: A Strategic Project Management Approach to the Study of Large-Scale Software Development Using System Dynamics," Massachusetts Institute of Technology , Tesis de Máster 2012.
- [6] Ventana System Inc. (2013) Vensim. [Online]. <http://www.vensim.com>
- [7] J Aracil, *Dinámica de Sistemas*. Madrid, España: Alianza Editorial, 1997.
- [8] Torrealdea J, *Dinámica de Sistemas. Elementos y Estructuras de un Modelo. Construyendo modelos.*: Ciencias de la Computación e Inteligencia Artificial. Universidad del País Vasco, 2003.
- [9] Diego Alberto Godoy, "Diseño de un Simulador Dinámico de Proyectos de Desarrollo de Software que utilizan Metodología Scrum," UNLP, La Plata, Tesis de Maestría 2014.
- [10] Diego Alberto Godoy, Edgardo A. Belloni, Henry Kotynski, Hector H Dos Santos, and Eduardo Omar Sosa, "Simulando Proyectos de Desarrollo de Software Administrados con Scrum," in *XVI Workshop de Investigadores en Ciencias de la Computación RedUNCI*, Ushuaia, 2014.
- [11] Diego Alberto Godoy, Eduardo Omar Sosa, Edgardo Belloni, and Henry Kotinski, "Simulación Dinámica de Gestión de Tareas en Proyectos Desarrollados Con Scrum," in *II Congreso Nacional de ingeniería informática/ingeniería de sistemas (CoNaIISI)*, Universidad Nacional de San Luis, San Luis, 2014, 2014.
- [12] Diego Alberto Godoy, Cristian Henry Kotyński, Edgardo Anibal Belloni, and Eduardo Omar Sosa, "Un Modelo de Simulación de Proyectos scrum con dinámica de sistemas.," in *WWW/INTERNET 2015 e COMPUTAÇÃO APLICADA 2015 FLORIANÓPOLIS*, Florianópolis, 2015, Organizada por IADIS International Association for Development of the Information Society.
- [13] Cristian Henry Kotyński, Diego Alberto Godoy, Edargardo Anibal Belloni, Eduardo Omar Sosa, and Matías Anibal Rodriguez, "Análisis de la Dinámica de los Recursos Humanos en proyectos Desarrollados Con Scrum mediante Dinámica de Sistemas.," , Ciudad Autónoma de Buenos Aires, 2015.