



Entorno de ingeniería de requisitos aplicado para producir software en una universidad

Environment to requirements engineering applied to produce software in an university

Lisbán Torres-Pérez^I, Martha Dunia Delgado-Dapena^{II}, Dailin Rodríguez-Nápoles^I, Darlene Gómez-Suárez^{II}, Wendy de la Torre-Parejo^I, Yasmína Alonso-Abreu^I

^I Complejo de Investigaciones Tecnológicas Integrada, CITI. La Habana, Cuba.

E-mail: ltorresp@udio.cujae.edu.cu, drodriguez@udio.cujae.edu.cu, wtorre@ceis.cujae.edu.cu, yalonso@ceis.cujae.edu.cu

^{II} Instituto Superior Politécnico José Antonio Echeverría, Cujae. Facultad de Ingeniería Informática. La Habana, Cuba.

E-mail: marta@ceis.cujae.edu.cu, dgomez@ceis.cujae.edu.cu

Recibido: 07/07/2011

Aprobado: 09/03/2013

RESUMEN

La ingeniería de requisitos es un tema de suma actualidad en las investigaciones a nivel mundial. Estudios actuales sobre el éxito de los proyectos de *software* reflejan que esta es una de las disciplinas, dentro del desarrollo de proyectos de *software*, que más incidencia tiene en el fracaso de los proyectos. En este trabajo se presentan como resultado las experiencias de aplicación de un entorno de ingeniería de requisitos para el desarrollo de proyectos de *software* en las condiciones de una universidad, en particular en el Instituto Superior Politécnico José Antonio Echeverría, Cujae. Dentro de esta universidad se aplicó el entorno en el Complejo de Investigaciones Tecnológicas Integradas (CITI) y se destacan las mejoras que se le incorporaron al mencionado entorno como parte del proceso de asimilación en la organización. Estos elementos se obtienen al aplicar una conjunción de métodos de investigación empíricos y teóricos que contribuyen a cumplimentar el objetivo de definir un entorno para la Ingeniería de Requisitos aplicable en el marco universitario.

Palabras clave: ingeniería de requisitos, gestión de requisitos, calidad de *software*, herramientas de gestión de requisitos.

ABSTRACT

Requirements Engineering is an outstanding topic in the investigations at world level. Current studies about the software projects success reflect that this it is one of the disciplines, inside the software projects development that more incidence has in the projects failure. In this work they are presented an environment to Requirements Engineering application experiences as a result for the software projects development at an university, particularly in the Instituto Superior Politécnico

José Antonio Echeverría, Cujae. Inside this university the environment was applied in the Complejo de Investigaciones Tecnológicas Integradas (CITI) and they stand out the improvements that were incorporated the one mentioned environment like part of the organization assimilation process. These elements are obtained when applying an empiric and theoretical investigation methods conjunction that contribute to execute the objective to define an environment for the Requirements Engineering applicable in the university mark.

Key words: requirements engineering, requirements management, software quality, requirements management tools.

I. INTRODUCCIÓN

La calidad de los productos de *software* está estrechamente relacionada con la calidad en el proceso de desarrollo de *software*. En particular la etapa de requisitos es reconocida como una etapa crítica para el éxito de los proyectos, puesto que la calidad del producto final está determinada, en gran medida, por la calidad del proceso de desarrollo en sí [1; 2; 3; 4]. El factor que tiene mayor impacto en el fracaso de un proyecto, es precisamente el tratamiento a los requisitos [5; 6]. Debido a la naturaleza cambiante de los requisitos, es necesario gestionarlos durante todo el desarrollo y explotación del *software*. La figura 1 muestra los factores de mayor impacto en el fracaso o éxito de los proyectos. Estos datos, tomados de una muestra de más de 170 mil proyectos de desarrollo de *software*, ilustran la gran incidencia de la especificación de requisitos [7; 8; 9].

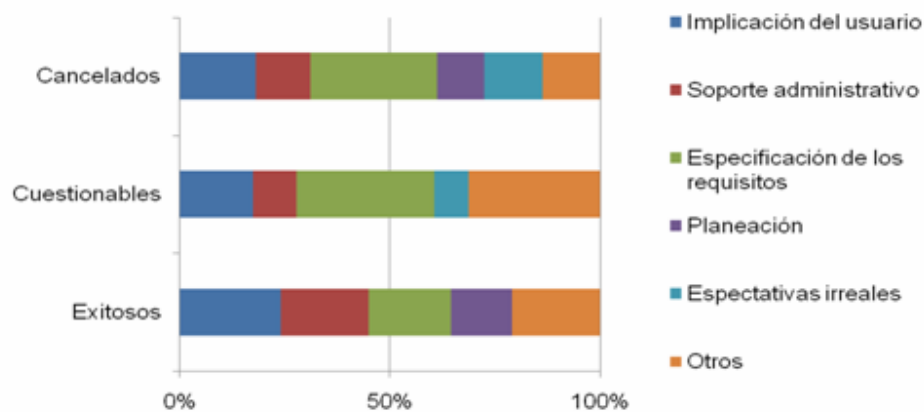


Figura 1. Factores que impactan el desarrollo de proyectos de *software*.
Fuente. Reporte Chaos, del StandishGroup (2010) [3].

En el entorno universitario la situación se complejiza aún más si se considera que la fuerza de trabajo está compuesta fundamentalmente por estudiantes y profesores. El Complejo de Investigaciones Tecnológicas Integradas (CITI), es una organización encargada de investigaciones técnicas dentro de un entorno universitario, donde la mayor parte de la fuerza laboral es a tiempo parcial; pues comparten las labores productivas con las responsabilidades docentes; esto implica que el personal no es constante, puesto que pueden graduarse y/o abandonar el equipo.

Los estudiantes en muchos casos no poseen experiencia e incluso en ocasiones no tienen todo el conocimiento requerido y forman parte de equipos de desarrollo de número reducido, en los cuales deben desempeñar varios roles, elemento este que retarda la especialización del personal en algunas tareas; es por ello que se impone la necesidad de tener guías precisas y útiles conformando un entorno para la aplicación de la Ingeniería de Requisitos en el proceso de desarrollo, de ahí el diseño de un procedimiento para tratar los requisitos de todos los proyectos de la organización [7; 10].

En la actualidad existen diversos y valiosos aportes a la Ingeniería de Requisitos [5; 6]. Múltiples autores abordan el tema desde diferentes perspectivas, desde los procesos, estándares y modelos, desde las actividades propias de la Ingeniería de Requisitos, o las técnicas a emplear en la captura de

ENTORNO DE INGENIERÍA DE REQUISITOS APLICADO PARA PRODUCIR SOFTWARE EN UNA UNIVERSIDAD

requisitos [11; 12; 13; 14]. Además debe señalarse como un factor fundamental las herramientas que dan soporte a los procesos [15; 16; 17; 18]. Estas propuestas están concebidas en su mayoría para ejecutarse en entornos empresariales donde la fuerza de trabajo está dedicada a esta única función [19]. Este aspecto no se ajusta al entorno de desarrollo de *software* en las universidades, por lo que se hace necesario hacer propuestas que respondan a las características de un centro universitario.

En el CITI se ha propuesto un Entorno para gestionar Requisitos de *Software* en los proyectos que se desarrollan, su implantación se documentó por Torres en [20]. Las experiencias de aplicar este entorno en un ambiente productivo enmarcado en la universidad: Instituto Superior Politécnico José Antonio Echeverría, Cujae.

II. MÉTODOS

El entorno está compuesto por un modelo de procesos, métodos y herramientas que dan soporte al proceso y que facilitan la actividad de gestión en este entorno universitario. Se elaboró para la Ingeniería de Requisitos, un flujo de trabajo que posee seis subprocesos, el cual se muestra en la figura 2. Cada subproceso contiene un conjunto de pasos, asociados a un conjunto de buenas prácticas para su ejecución. Como parte de la propuesta se definieron plantillas para cada uno de los artefactos que la necesitaban, se elaboraron guías de construcción y 8 listas de chequeo que ayudan en su confección y evaluación [21; 22; 23]. Este procedimiento reduce la cantidad de actividades y artefactos propuestos en RUP, aunque sin perder la esencia del proceso y el propósito perseguido, elemento este significativo, pues está acorde con los lineamientos de la organización, que persigue agilizar los procesos de desarrollo sin que se pierdan las necesidades básicas, manteniendo su calidad [19].

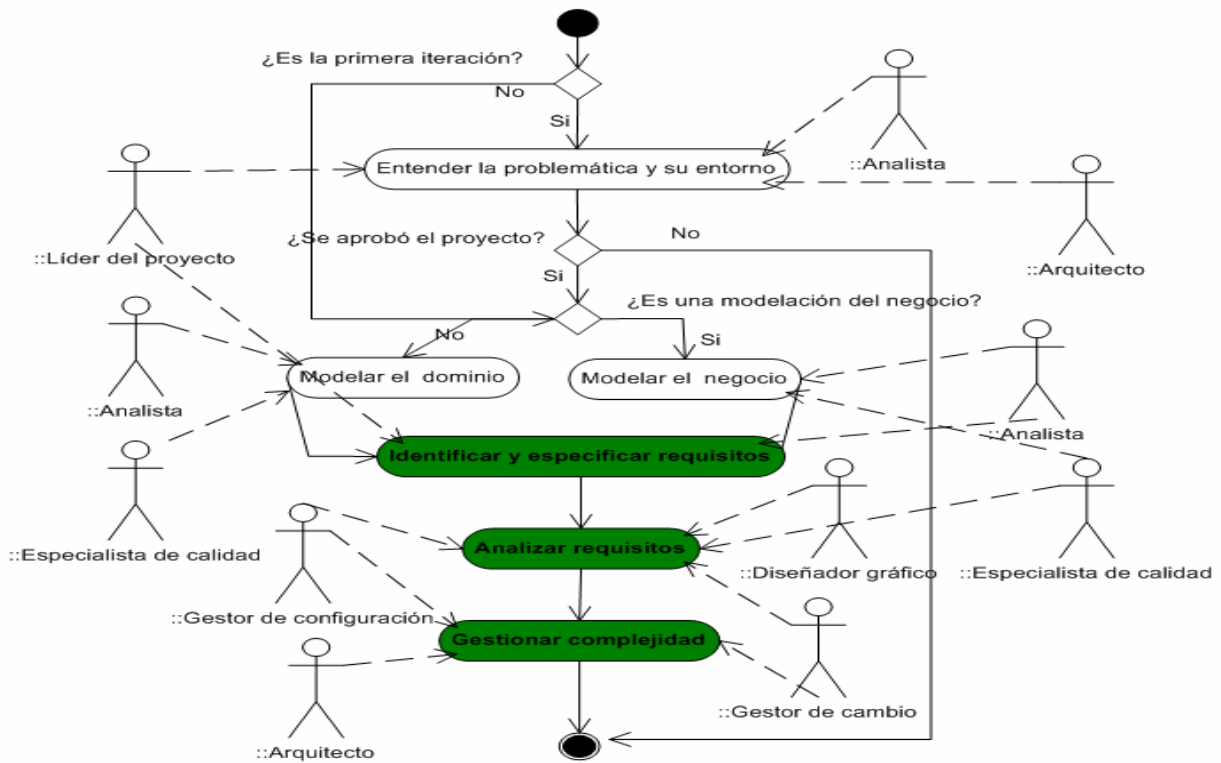


Figura 2. Flujo de trabajo para la aplicación de la Ingeniería de Requisitos.

III. RESULTADOS

El proceso de validación se tradujo en una mejora continua de las actividades, los roles y los artefactos en lo particular y del entorno de Ingeniería de Requisitos en general. Además ha permitido la recogida de datos relevantes que ilustran como se refinó el producto final, que puede ser empleado en el futuro por quienes necesiten desarrollar uno similar, no solo desde el punto de vista de los resultados concretos, sino también por los métodos empleados.

El subproceso que se muestra en la figura 3 tiene como propósito comprender el estado actual de la entidad para la cual la aplicación va a ser desarrollada; en términos de su actual proceso, las herramientas, actitudes de las personas, clientes, tendencias, problemas técnicos y áreas de mejora.

ENTORNO DE INGENIERÍA DE REQUISITOS APLICADO PARA PRODUCIR SOFTWARE EN UNA UNIVERSIDAD

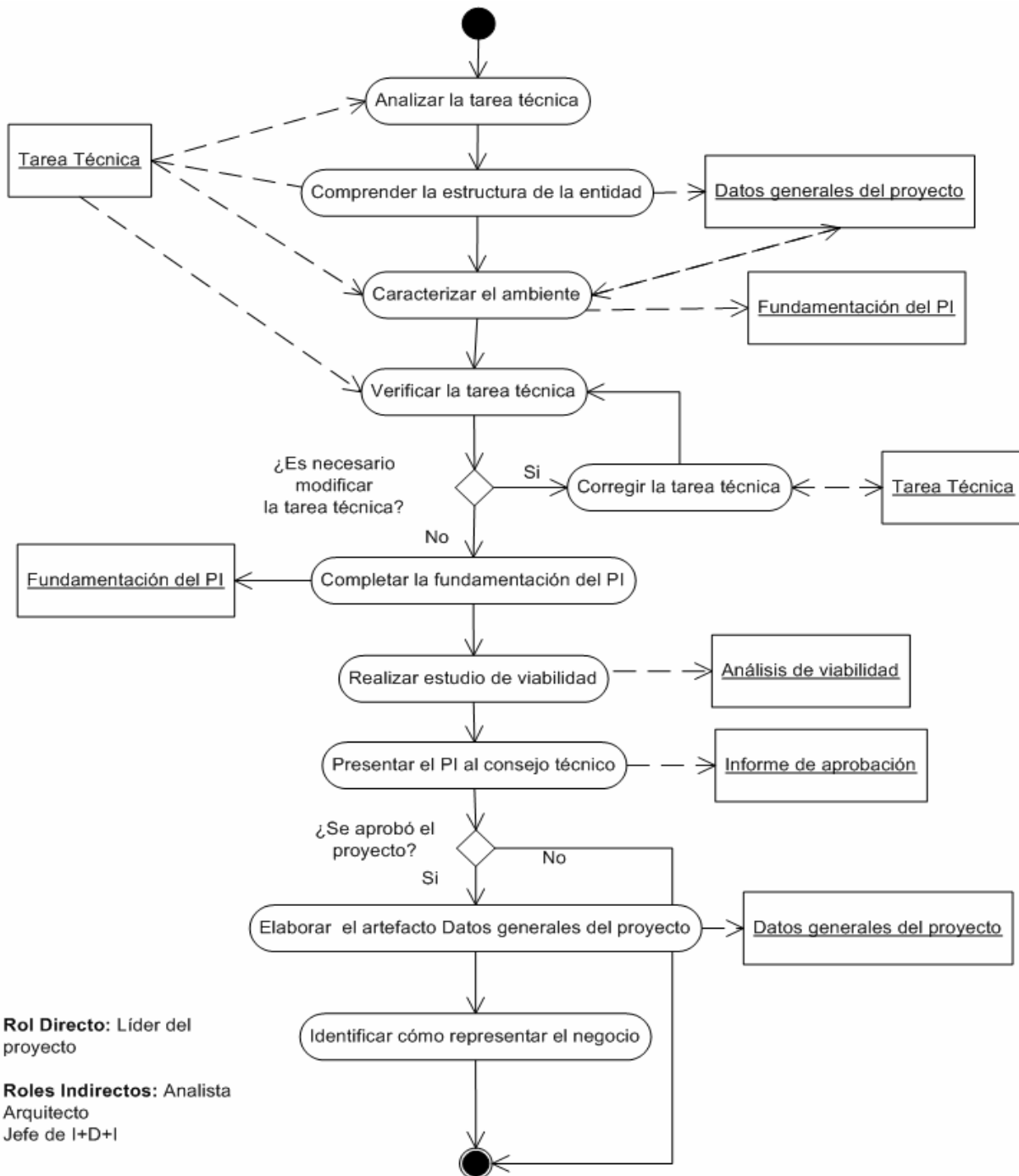


Figura 3. Flujo de trabajo del Subproceso "Entender la problemática y su entorno".

En el subproceso se confecciona la información necesaria para la aprobación final del Proyecto de Investigación (PI) y se identifica la participación de los clientes en el proceso.

El subproceso que se muestra en la figura 4 tiene el propósito de describir las actividades del negocio, la información de flujo a través de ella, los orígenes y destinos de la información; las actividades específicas son detalladas por Torres (2011) [20].

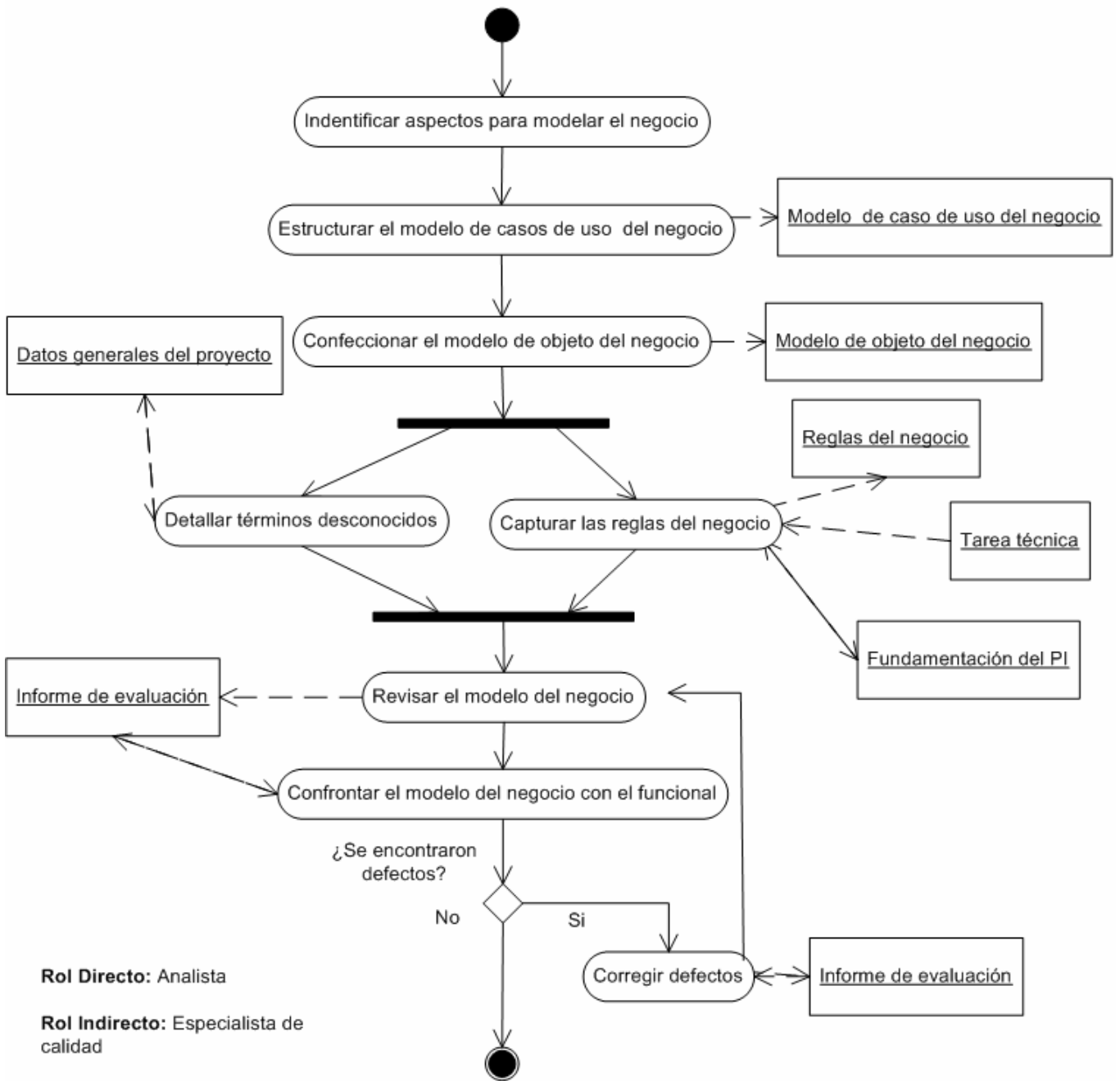


Figura 4. Flujo de trabajo del Subproceso "Modelar el negocio".

Como resultado del proceso de pilotaje surgieron mejoras que se realizaron al entorno, la clasificación de estas mejoras se muestran en las siguientes figuras se muestra en la figura 5:

ENTORNO DE INGENIERÍA DE REQUISITOS APLICADO PARA PRODUCIR SOFTWARE EN UNA UNIVERSIDAD

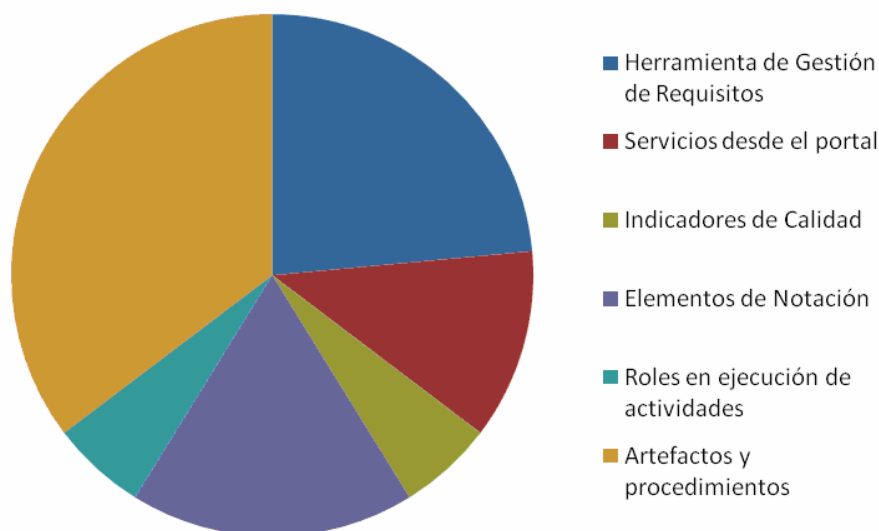


Figura 5. Clasificación de Mejoras propuestas durante el pilotaje.

Un aspecto importante dentro de este proceso lo constituyó por una parte, la creación de listas de chequeo para evaluar el entendimiento y la satisfacción del personal de los proyectos pilotos con los diferentes componentes del entorno (artefactos, actividades y disciplinas), y por otra parte encuestas, con un objetivo común a las primeras.

Un acápite notorio de la aplicación del entorno, fue la herramienta, que tras su evaluación por los participantes donde solo un encuestado consideró que la herramienta no contribuye notablemente, basándose en que es genérica y que no se ajusta con exactitud a sus necesidades. El resto de los participantes determinó que la herramienta apoya de forma elevada al proceso de gestión de los requisitos.

Para lograr el éxito de la investigación, teniendo como resultado principal implantar un entorno de Ingeniería de Requisitos de *software* en el CITI, fue fundamental contar con el apoyo de la dirección de la organización y de los proyectos que han participado en el pilotaje.

Las características de los proyectos seleccionados en cuanto a composición de sus requisitos se muestran en la tabla 1.

Tabla 1. Características de los proyectos pilotos.

Proyecto	Requisitos Iniciales	Requisitos Funcionales	Requisitos No Funcionales
Proyecto 1	5	19	4
Proyecto 2	20	85	0
Proyecto 3	11	18	2
Proyecto 4	3	1	0
Proyecto 5	6	19	2
Total	45	142	8

El entorno que se diseñó, se nutrió de los principios planteados por CMMI en su área de procesos de administración de requisitos [24]. Es importante resaltar el hecho de que se ha alcanzado un grado de acercamiento en los aspectos fundamentales con este modelo. A continuación se muestra por cada práctica específica (SP, por sus siglas en inglés), subpráctica y evidencia sugerida el nivel de cumplimiento en el entorno propuesto, que impactan positivamente en la organización enfrascada en la actualidad en un programa de mejora guiado por este modelo.

Práctica Específica 1: Obtener el entendimiento de los requisitos.

En la tabla 2 se muestra el cumplimiento de la SP “Obtener el entendimiento de los requisitos” y en la tabla 3 se muestran las evidencias sugeridas para ello.

Tabla 2. Cumplimiento de la SP “Obtener el entendimiento de los requisitos”.

Subpráctica	Descripción
Establecer criterios para evaluar a los proveedores de requisitos.	Se cumple, pues como parte del programa de mejoras de la organización, se diseñó una lista de criterios ponderados para la selección de los proveedores de requisitos.
Establecer criterios objetivos para evaluar y aceptar requisitos.	Están definidas listas de chequeo para la evaluación de los requisitos del cliente y del producto.
Analizar los requisitos para asegurar que se cumple con los criterios establecidos.	Existen las actividades para Analizar requisitos, con guías de técnicas para la ejecución de estas, y registrar los defectos, encontrados un artefacto que se diseñó para ello.
Lograr con los proveedores de requisitos un entendimiento de los requisitos, de tal forma que los participantes del proyecto puedan comprometerse con ellos.	Existen pasos para confrontar con los interesados los requisitos especificados y de estas reuniones se confecciona un acta con los resultados.

Tabla 3. Evidencias de la SP “Obtener el entendimiento de los requisitos”.

Evidencia sugerida	Cumplimiento
Lista de criterios para distinguir los proveedores apropiados de requisitos.	Si
Criterios para la evaluación y aceptación de los requisitos.	Si
Resultados de análisis contra los criterios.	Si
Conjunto de requisitos aceptados.	Si

Práctica Específica 2: Obtener el compromiso a los requisitos.

En la tabla 4 se muestra el cumplimiento de la SP “Obtener el compromiso con los requisitos” y en la tabla 5 se muestran las evidencias sugeridas para ello.

ENTORNO DE INGENIERÍA DE REQUISITOS APLICADO PARA PRODUCIR SOFTWARE EN UNA UNIVERSIDAD

Tabla 4. Cumplimiento de la SP "Obtener el compromiso a los requisitos".

Subpráctica	Descripción
Evaluar el impacto de los requisitos en los compromisos existentes.	Los requisitos se documentan con un tiempo estimado en horas/hombres que permite evaluar su impacto en los compromisos planificados.
Negociar y registrar los compromisos.	Existen pasos definidos para la negociación con los interesados que deben firmar los requisitos especificados por el equipo de proyecto.

Tabla 5. Evidencias de la SP "Obtener el compromiso a los requisitos".

Subpráctica	Descripción
Evaluar el impacto de los requisitos en los compromisos existentes.	Los requisitos se documentan con un tiempo estimado en horas/hombres que permite evaluar su impacto en los compromisos planificados.
Negociar y registrar los compromisos.	Existen pasos definidos para la negociación con los interesados que deben firmar los requisitos especificados por el equipo de proyecto.

Práctica Específica 3: Administrar los cambios a los requisitos.

En la tabla 6 se muestra el cumplimiento de la SP "Administrar los cambios a requisitos" y en la tabla 7 se muestran las evidencias sugeridas para ello.

Tabla 6. Cumplimiento de la SP "Administrar los cambios a los requisitos".

Subpráctica	Descripción
Documentar todos los requisitos y los cambios a los requisitos generados por el proyecto.	Se utiliza una herramienta de gestión (Jama Contour) que garantiza esta funcionalidad de forma automática. Las otras herramientas destacadas (OSRMT y TFS) también cumplen esta funcionalidad.
Mantener un historial de los cambios a los requisitos con su justificación.	La herramienta de gestión mantiene un historial de cambios con el versionado de cada uno de los cambios realizados.
Evaluar el impacto de los cambios a los requisitos desde el punto de vista de los involucrados relevantes.	Las herramientas destacadas en la propuesta apoyan este proceso y además se documentan los impactos de un cambio durante su análisis en diferentes áreas del proyecto como pueden ser la arquitectura, la seguridad, la documentación y los recursos.
Mantener disponibles los datos de los requisitos y de sus cambios para el proyecto.	Funcionalidad que garantiza la herramienta utilizada de forma automática para los usuarios con permisos sobre el proyecto.

Tabla 7. Evidencias de la SP “Administrar los cambios a los requisitos”.

Evidencia sugerida	Cumplimiento
Estado de los requisitos.	Si
Base de datos de los requisitos.	Si
Base de datos de las decisiones tomadas con los requisitos.	Si

Práctica Específica 4: Mantener la traza bidireccional de los requisitos.

En la tabla 8 se muestra el cumplimiento de la SP “Mantener la traza bidireccional de los requisitos” y en la tabla 9 se muestran las evidencias sugeridas para ello.

Tabla 8. Cumplimiento de la SP “Mantener la traza bidireccional de los requisitos”.

Subpráctica	Descripción
Mantener el trazo de los requisitos para asegurar que la fuente de los requisitos de menor nivel (derivados) esté documentada.	Funcionalidad que brindan cada una de las herramientas destacadas a partir del registro de las relaciones entre los requisitos tanto del cliente como del producto.
Mantener el trazo desde los requisitos originales hasta sus requisitos derivados y en la asignación de sus funciones, interfaces, objetos, personas, procesos y productos de trabajo.	Funcionalidad brindada por las herramientas destacadas a partir del registro de las relaciones correspondientes.
Generar la matriz de trazabilidad de los requisitos.	Funcionalidad que realizan las herramientas destacadas automáticamente a partir del registro de las relaciones correspondientes.

Tabla 9. Evidencias de la SP “Mantener la traza bidireccional de los requisitos”.

Evidencia sugerida	Cumplimiento
Matriz de trazo de requisitos.	Si
Sistema de trazo de requisitos.	Si

Práctica Específica 5: Identificar inconsistencias entre el trabajo del proyecto y los requisitos.

En la tabla 10 se muestra el cumplimiento de la SP “Identificar inconsistencias entre el trabajo del proyecto y los requisitos” y en la tabla 11 se muestran las evidencias sugeridas para ello.

ENTORNO DE INGENIERÍA DE REQUISITOS APLICADO PARA PRODUCIR SOFTWARE EN UNA UNIVERSIDAD

Tabla 10. Cumplimiento de la SP "Identificar inconsistencia entre el trabajo del proyecto y los requisitos".

Subpráctica	Descripción
Revisar los planes del proyecto, las actividades y los productos de trabajo para detectar inconsistencias con los requisitos y con sus cambios.	Se diseñó como parte de la gestión de requisitos un subproceso donde se revisan los productos de trabajo para detectar las inconsistencias.
Identificar la fuente de la inconsistencia y su razonamiento.	Los productos de trabajo se verifican, como parte del flujo de trabajo, y sus inconsistencias son registradas en un artefacto diseñado para ello. Estos datos son obligatorios en la herramienta de gestión de requisitos.
Identificar los cambios a los planes y productos de trabajo que requieren realizarse como resultado de los cambios a los requisitos de la línea base.	El subproceso de control de inconsistencias contiene actividades que establecen el tratamiento de todos los cambios y las actualizaciones de los planes y productos.
Iniciar las acciones correctivas.	El subproceso de inconsistencias posee actividades para determinar acciones correctivas que son seguidas.

Tabla 11. Evidencias de la SP "Identificar inconsistencia entre el trabajo del proyecto y los requisitos".

Evidencia sugerida	Cumplimiento
Documentación de las inconsistencias, que incluyan las fuentes, las condiciones y el razonamiento.	Si
Acciones correctivas.	Si

El análisis expuesto anteriormente, evidencia un elevado nivel de cumplimiento de los principios de CMMI; dando una muestra de su calidad en el CITI. Se considera relevante señalar algunos aspectos que han contribuido a la mejora del proceso de ingeniería de requisitos en la entidad:

- Los proyectos piloteados formalizaron la especificación del total de sus requisitos.
- Se han detectado y corregido errores en las especificaciones de requisitos.
- Se determinaron como las causas fundamentales de los defectos en los requisitos la ambigüedad, el solapamiento y la omisión de requisitos considerados como obvios por los proveedores.
- Se logró registrar la trazabilidad bidireccional de los requisitos con los productos obtenidos en el trabajo de los proyectos, detectándose y corrigiéndose varias inconsistencias que hubiesen incrementado la probabilidad de futuros errores.
- La formalización adecuada de los requisitos permitió aplicar técnicas para clasificarlos y comenzar a trabajar en un repositorio de los requisitos más empleados en los proyectos del CITI, que estarían relacionados con los productos que de ellos se obtienen; este aspecto debe potenciar el re-uso de la experiencia y los productos en la organización.
- Se obtuvo una guía para capturar los requisitos que hace énfasis en los principales problemas detectados en el pilotaje.
- Se obtuvo una guía para ayudar a revisores inexpertos a detectar inconsistencias entre los requisitos y los productos de trabajo del proyecto.
- Se obtuvo una guía para evaluar el impacto de los cambios para los compromisos del proyecto, que contribuye a favorecer la toma de decisiones y la interacción con los clientes, teniendo en cuenta la naturaleza cambiante de los requisitos.
- Se familiarizó y concientizó a los miembros de los proyectos con la aplicación del entorno para la ingeniería de requisitos.
- Se documentó una guía para el proceso, favoreciendo que pueda ser seguido por personal con poca experiencia en el tema de la ingeniería de requisitos; este aspecto contribuye a mitigar los riesgos asociados a las características del personal que labora en el CITI.

IV. DISCUSIÓN

Para aplicar el Entorno de Gestión de Requisitos para el CITI, se decidió hacerlo desde dos perspectivas, una primera que será pilotaje de un grupo de proyectos que se desarrollan en el CITI y una segunda comparando la alineación del procedimiento definido con el área de Administración de Requisitos del Modelo de Madurez de Capacidades de *Software* Integrado –conocido por sus siglas en inglés CMMi- que permita continuar el proceso de mejora del entorno definido.

Para validar el entorno se decidió hacer un primer pilotaje en cinco proyectos que se ejecutan en el CITI. Estos proyectos seleccionados de líneas de investigación diferentes, siguieron el proceso propuesto y utilizaron la herramienta de soporte. Este pilotaje tuvo dos objetivos fundamentales. El primero la medición de la aceptación del entorno y su usabilidad. El segundo la mejora del entorno propuesto a partir de las sugerencias y dificultades surgidas durante su aplicación en los proyectos pilotos.

ENTORNO DE INGENIERÍA DE REQUISITOS APLICADO PARA PRODUCIR SOFTWARE EN UNA UNIVERSIDAD

Conjuntamente con la selección de los proyectos se definieron actividades para la validación del entorno que se exponen a continuación:

- Capacitación de los equipos de desarrollo de *software* de cada proyecto sobre el nuevo entorno propuesto, tanto en los procedimientos a seguir como en las herramientas a utilizar. Se desarrollaron dos seminarios para preparar al personal involucrado en el pilotaje.
- Creación de un sitio con el objetivo de mantener la información socializada. Todos los involucrados en el pilotaje podían mantenerse al tanto de cada detalle, tanto de la secuencia de actividades y los resultados parciales del proceso de validación, como el registro de los criterios del personal de los proyectos pilotos.
- Celebración de 10 talleres de soporte, en semanas alternas, donde el personal que propuso el entorno, se encontraba con el resto de los implicados en el pilotaje, para aclarar inquietudes, que eran registradas y analizadas como parte del proceso de retroalimentación. Estos talleres se finalizaban con las lecciones aprendidas y las propuestas de mejora al entorno.
- Incorporación de mejoras al proceso.
- Aplicación de encuestas a los equipos de proyectos sobre la facilidad de uso del entorno propuesto y su aceptación. Cada una de las sugerencias que se obtuvieron a través de las encuestas fueron discutidas en los talleres para aprobar o no, la introducción en el entorno de las mejoras correspondientes.

VI. CONCLUSIONES

Este artículo plantea los resultados parciales de un proceso de asimilación y aplicación de un entorno propio del CITI para la Ingeniería de Requisitos en el desarrollo de Proyectos de *Software*. Este proceso no está terminado aún y sigue perfeccionándose, pero debe destacarse hasta este momento:

1. Obtención de un conjunto de experiencias y mejoras en el entorno que permiten tener una vía para gestionar los requisitos de los proyectos de *software* y que facilita la ejecución de este proceso en las condiciones particulares de desarrollo de *software* en el CITI.
2. Creación de una conciencia en los equipos de desarrollo participantes del pilotaje sobre la manera de realizar una correcta Ingeniería de requisitos
3. Alineación del entorno con los planteamientos del modelo de referencias CMMI para la gestión de requisitos.

En el futuro deberá valorarse si esta experiencia puede ser extendida a otros centros que desarrollen *software* en el marco de las universidades y con características similares en la composición de sus equipos de desarrollo de *software*. 🏠

VI. REFERENCIAS

1. BASTANI, B., «A Requirements Analysis Framework for Open Systems Requirements Engineering», *ACM SIGSOFT Software Engineering Notes* [en línea], 2007, vol. 32, no. 2, pp. 1-19 [consulta: 23/12/2012], ISSN 0163-5948. Disponible en: <<http://dl.acm.org/citation.cfm?doid=1234741.1234753>>
2. CHENG, B., «Research Directions in Requirements Engineering», en *Future of Software Engineering(FOSE'07)* California, 2007, [consulta: ISBN 0-7695-2829-5. Disponible en:
3. DAMIAN, D.; CHISAN, J., «An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that Lead to Payoffs in Productivity, Quality, and Risk Management», *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING* [en línea], 2006, vol. 32, no. 7, pp. 433-453 [consulta: 13-11-2013], ISSN 0098-5589. Disponible en: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1677531&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D1677531>
4. STANDISHGROUP, «Chaos Report», *Standish Group Internal Report* [en línea], 2010, [consulta: Disponible en: <<http://www.csus.edu/indiv/v/velianitis/.../ChaosReport.pdf>>

L. TORRES-PÉREZ, M. DELGADO-DAPENA, D. RODRÍGUEZ-NÁPOLES, D. GÓMEZ-SUÁREZ, W. DE LA TORRE-PAREJO

5. DUBE, R.; RAJENDRA, D., «Process-oriented Complete Requirement Engineering Cycle for Generic Projects», en *International Conference and Workshop on Emerging Trends in Technology (ICWET 2010)* Mumbai, 2010, [consulta: ISBN 978-1-60558-812-4. Disponible en:
6. FAGAN, M., *Design and code inspections to reduce errors in program development*, New York, Springer-Verlag 2002, ISBN 3-540-43081-4, pp. 575-607.
7. IEEE, «Standard Glossary of Software Engineering Terminology», *IEEE - 610* [en línea], 1990, pp. 12-17 [consulta: 12/12/2012], Disponible en: <www.standards.ieee.org>
8. MICH, L.; GARIGLIANO, R., «NL-OOPS: A requirements analysis tool based on natural language processing», en *3rd International Conference on Data Mining 2002* ciudad, 2002, vol. 3, pp. 321-330. [consulta: ISSN 1-85312-925-9. Disponible en:
9. PACHECO, A., «La Identificación de Stakeholders en la Ingeniería de Requisitos», en *Proc. of the 8th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2009* 2009, [consulta: ISSN 978-0-88986-641-6. Disponible en:
10. ZAVE, P., «Four dark corners of requirements engineering», *ACM Transactions on Software Engineering and Methodology* [en línea], 1997, 6, 1, pp. 1-30 [consulta: 18-05-2012], ISSN 1049-331X. Disponible en: <<http://www.google.com/cu/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CC4QFjAA&url=http%3A%2F%2Fdl.acm.org%2Fcitation.cfm%3Fid%3D237434&ei=ht3eUpztFYW0ygG-vIHQAQ&usq=AFQjCNHJWRzFB3aHV8tHcP1Zpzza3XskYw&bvm=bv.59568121,d.aWc>>
11. FINKELSTEIN, A., «Abstraction-based requirements management», en *International workshop on Role of abstraction in software engineering* Shanghai, China, 2006 pp. 3-10. [consulta: ISBN 0-7803-9593-X. Disponible en:
12. FRITZINGER, E., «Support tool for the organization of requirements modeling», *Computer Science* [en línea], 2006, [consulta: Disponible en: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4017708&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D4017708>
13. GLASS, R., *Software Engineering*, New York, Addison Wesley, 2002, ISBN 0321117425.
14. HAMLET, D., «Tools and Experiments Supporting a Testing-Based Theory of Component Composition» *ACM Transactions on Software Engineering and Methodology*, 2009, vol. 18, no. 3, pp. 1-41, ISSN 1049-331X.
15. HRVOJE, Ž.; ANTUN, C., «RUP-Based Process Model for Security Requirements Engineering in Value-Added Service Development Software Engineering for Secure Systems », en *SESS'09* Vancouver, 2009, [consulta: ISBN 978-1-4244-3725-2 Disponible en:
16. MATEO, P.; SEVILLA, D., «Open HMI Tester: un Framework Open-source para herramientas de pruebas de software» *Taller de la Jornada de Ingeniería del Software y Bases de Datos*, 2009, vol. 3, no. 4, pp. 1-12, ISSN 1988-3455.
17. MCDONALD, B., «Definición de Perfiles en Herramientas de Gestión de Requisitos», [tesis de doctorado], Madrid, Universidad Politécnica de Madrid, Departamento de Lenguajes y Sistemas Informáticos e Ingeniería del Software, 2005.
18. SEYFF, N.; MAIDEN, N.; TOSAR, A., «Requirements Engineering Tools Go Mobile», en *26th International Conference on Software Engineering (ICSE'04)* Scotland, UK, 2004, [consulta: ISBN 0-7695-2163-0. Disponible en:
19. PMI, *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, 4ta. ed., New York, Barnes & Noble, 2009, ISBN 978-1933890517, pp. 459.
20. TORRES, L., «Definición de un entorno para la aplicación de la ingeniería de requisitos en el CITI», [tesis de maestría], La Habana, Instituto Superior Politécnico José Antonio Echeverría, Cujae, Departamento de ingeniería de software, 2011.
21. PIATTINI, M., *Fábricas de Software: experiencias, tecnologías y organización*, 2da. ed., ciudad, RA-MA Editorial, 2007, ISBN 978-84-7897-809-0, pp. 555.
22. WAN, J.; WAN, D., «Research on Knowledge Creation in Software Requirement Development Software Engineering & Applications», [en línea], 2010, pp. 487-494 [consulta: 13/09/2012], Disponible en: <www.scirp.org/journal/PaperDownload.aspx?PaperID=1753>

ENTORNO DE INGENIERÍA DE REQUISITOS APLICADO PARA PRODUCIR SOFTWARE EN UNA UNIVERSIDAD

23. YANN-GAE, G.; LAURENCE, D.; ANNE-FRANCOISE, L., «DECOR: A Method for the Specification and Detection of Code and Design Smells» *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, 2010, vol. 36, no. 1, pp. 20-36, ISSN 0098-5589.
24. SOMMERVILLE, I., «An Empirical Study of Industrial Requirements Engineering Process Assessment and Improvement», *ACM Transactions on Software Engineering and Methodology* [en línea], 2005, vol. 14, no. 1, pp. 85-117 [consulta: 12/12/2012], ISSN 1049-331X. Disponible en: <www.dcs.bbk.ac.uk/~sven/tmp/p85-sommerville.pdf>