

# **Técnicas de Ingeniería de Requerimientos aplicadas en proyectos de desarrollo de software y su relación con la investigación y la docencia universitaria. Un caso de estudio.**

Sonia I. Mariño, María V. Godoy, Pedro Alfonso, Alejandra Matoso, Mirta Fernandez

Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura. 9 de Julio 1449. CP: 3400. Corrientes. Argentina. Universidad Nacional del Nordeste.  
[simarinio@yahoo.com](mailto:simarinio@yahoo.com), [mvgodoy@exa.unne.edu.ar](mailto:mvgodoy@exa.unne.edu.ar)

**Resumen.** Se sintetiza un estudio elaborado al interior de un equipo de trabajo de I+D, docentes de la carreras Licenciatura en Sistemas de Información, en referencia a la aplicación de conceptos comprendidos en los distintos núcleos de la Ingeniería de Requerimientos, según lo expuesto por ACM para la curricula 2013. El trabajo logrado permite reflexionar respecto al nivel de aplicabilidad de estos conceptos en la generación de artefactos de software. El objetivo es continuar avanzando y fortaleciendo articulaciones entre actividades de docencia e I+D, donde algunos aspectos detectados constituirán información a priori para proponer acciones posteriores, encaminadas a la mejora de la calidad de la enseñanza universitaria.

**Palabras clave:** I+D, Ingeniería del Software, núcleos básicos de Informática Ingeniería de Requerimientos, plan curricular

## **1 Introducción**

La Ingeniería del Software emplea métodos ingenieriles, procesos, técnicas y medidas. Siguiendo a ACM [1], se beneficia de la utilización de herramientas para la gestión de desarrollo de software, el análisis y modelado de artefactos de software, la evaluación y control de calidad, y asegura un enfoque disciplinado y controlado a la evolución del software y su reutilización

Siguiendo lo expuesto por la Red UNCI, las tendencias mundiales en este sentido, además de los estándares de Argentina, otros estándares y los documentos internacionales como ACM [1] y IEEE [8], se trabaja permanentemente en el seno de las Universidades en revisiones de los perfiles establecidos, analizando las cargas horarias de las diversas titulaciones y las horas por trayecto curricular.

En la carrera Licenciatura en Sistemas de Información de la FaCENA, UNNE, actualmente según los programas y planes vigentes, se aborda dicha temática, la Ingeniería de Requerimientos en de las asignaturas “Ingeniería del Software I” e “Ingeniería del Software II”.

En [10] se define la Ingeniería de Requerimientos (IR) como la sistematización aplicada en el desarrollo de los mismos, a través de un proceso iterativo y cooperativo que consiste en analizar el problema, documentar los resultados en una variedad de formatos y probar la exactitud del conocimiento adquirido. Además, clasifica los procesos de IR en: i) elicitación, ii) especificación y iii) validación. Siguiendo a [16] la “Ingeniería de Requerimientos ayuda a los ingenieros de software a entender mejor el problema en cuya solución trabajarán. Incluye el conjunto de tareas que conducen a comprender cuál será el impacto del software sobre el negocio, qué es lo que el cliente quiere y cómo interactuarán los usuarios finales con el software” En [19] expresa que la Ingeniería de Requerimientos “es el proceso de desarrollar una especificación de software. Las especificaciones pretenden comunicar las necesidades del sistema del cliente a los desarrolladores del sistema” [19].

En la IR, una etapa crítica en el proceso de desarrollo de software, son los errores cometidos, originando problemas que se ponen de manifiesto en el diseño e implementación del sistema.

En relación a la especificación de requerimientos del software o SRS, la IEEE [8] la define como una descripción para un producto software particular, programa o conjunto de programas, que realizan ciertas funciones en un ambiente específico. Esta debe explicitar aquellas que van a ser realizadas, en qué situación y para quién, los resultados a obtener, como así también centrarse en los servicios a llevar a cabo.

Los requerimientos pueden dividirse en funcionales (describen lo que el sistema debe hacer y dependen del tipo de software a desarrollarse) y no funcionales, es decir las propiedades emergentes del sistema como la fiabilidad, el tiempo de respuesta, entre otros.

Algunos de estos factores pueden restringir el proceso que se debe utilizar para desarrollar el sistema. Un ejemplo de requerimientos de procesos es la especificación de los estándares de calidad que se deben utilizar [19].

Por otra parte, la Universidad como institución educativa está dedicada a formar recursos humanos profesionales que aportan a diversas áreas de la sociedad, caracterizándose por una mirada universal y abierta a variadas maneras de producir los conocimientos. Desde sus ámbitos se abordan funciones sustantivas: docencia, investigación, extensión y transferencia. Existe un consenso actual en evitar fragmentaciones entre las mismas, intentando articular sus acciones y resultados en pro del crecimiento y aplicación de los conocimientos, generados en los diversos espacios, para la resolución y atención de problemas abstraídos del mundo real. La sociedad actual no es viable imaginarla sin relacionarla fuertemente con la investigación, como motor de avance de la tecnología y de la informática.

En este trabajo, se presentan los avances vinculados a la revisión, estudio y aplicación de la IR y las actividades I+D derivadas y enmarcadas en dos proyectos acreditados por la Secretaría General de Ciencia y Técnica de la Universidad Nacional del Nordeste. Es decir, se considera relevante la aplicación de procesos y técnicas de IR, para la generación de artefactos software de calidad. Además, se enfatiza el estudio de estas técnicas en las diferentes metodologías de desarrollo.

Se aborda una descripción de los distintos núcleos de la Ingeniería de Requerimientos, comprendida en la Ingeniería del Software, según lo expuesto por ACM [1] a fin de evaluar el nivel de aplicabilidad en el equipo de trabajo de I+D, algunos de cuyos integrantes se desempeñan como docentes de la carrera Licenciatura

en Sistemas de Información. Por su parte, la Red UNCI [21], concluyó que el crecimiento de la disciplina, incrementa los fundamentos que deben incorporarse en una futura curricula o que la misma debe tender a la planificación de asignaturas obligatorias y electivas.

## 2 Materiales y Métodos

En el marco del proyecto, en la línea de I+D referente a IR, se efectuó:

- La revisión de metodologías de desarrollo de software. Se mencionan el Proceso Unificado de Rational (RUP) [18], el Proceso Unificado de Desarrollo del Software [9] y SCRUM [5].
- La elección, análisis y estudio de herramientas informáticas y su utilización para la representación y explicitación de los casos de estudios abordados. Se nombran: ArgoUML [3], BOUML [6] o STAR UML [20].  
En este trabajo, la metodología adoptada constó de las siguientes fases:
- La selección de contenidos del Núcleo Ingeniería de Requerimientos comprendida en la Ingeniería del Software y sus objetivos de aprendizaje [1].
- El diseño de una encuesta, que fue completada por los integrantes del equipo de trabajo del proyecto. La muestra constó de 9 participantes. Se requirió información vinculada: i) al conocimiento y aplicación de los contenidos definidos en los tres niveles del Núcleo de IR, ii) al cumplimiento de los objetivos expuestos en el documento.
- Sistematización de la información relevada y análisis de los resultados.

Se utilizaron como fuentes primarias de datos el documento ACM [1], la información elicitada de los recursos humanos docentes-investigadores en formación que integran el equipo de I+D orientado a la producción y transferencia de soluciones software al medio.

## 3 Resultados

Se coincide con ACM [1] que en general los estudiantes aprenden mejor a nivel de aplicación y participando e integrando un proyecto de desarrollo de software. Así, sus miembros trabajan en equipo involucrándose en el mayor número de fases del ciclo de vida.

Una de las metas de un proyecto informático de I+D es la modelización y construcción de artefactos software, enfatizando la transferencia de tecnología aplicando las teorías y metodologías de la Ingeniería de Software. La incorporación de recursos humanos de grado y postgrado ha permitido avanzar en esta línea temática concretándose los desarrollos que se mencionan en [2], [4], [7], [11], [12], [13], [14], [15], [17].

En referencia a la IR, particularmente en el modelado de artefactos software, en concordancia por lo expuesto por ACM [1], se detallan en las Tablas 1 a 3 los contenidos tratados y su aplicación en el equipo de trabajo.

En las Tablas 1 a 3 y 5 a 7 se especifica el número de miembros del equipo de I+D que respondieron positivamente, negativamente o no contestaron el cuestionario realizado sobre los Contenidos electivos del Núcleo de los Niveles 1 y 2 y del Nivel electivo y los resultados de aprendizaje de los correspondientes niveles.

**Tabla 1.** Contenidos electivos del Nucleo de Nivel 1

<b>Contenidos electivos del Núcleo de Nivel 1</b>	<b>SI</b>	<b>NO</b>
Fundamentos de la elicitación de requerimientos y de modelado	9	

**Tabla 2** Contenidos electivos del Núcleo de Nivel 2

<b>Contenidos electivos del Núcleo de Nivel 2</b>	<b>SI</b>	<b>NO</b>	<b>NC</b>
Propiedades de los requisitos, incluyendo la consistencia, validez, integridad y viabilidad	9		
Elicitación de requerimientos de software	5	4	
Descripción de los requisitos funcionales, utilizando, por ejemplo, casos de uso o las historias de los usuarios	9		
Los requerimientos no funcionales y su relación con la calidad del software	1	2	6
Descripción de los datos del sistema, utilizando, por ejemplo, los diagramas de clases o diagramas de entidad-relación	9		
La evaluación y el uso de especificaciones de requerimientos	9		

**Tabla 3.** Contenidos del Núcleo de Nivel electivo

<b>Contenidos del Núcleo de Nivel electivo</b>	<b>SI</b>	<b>NO</b>	<b>NC</b>
Técnicas de análisis y modelado de requerimientos	9		
La aceptabilidad de las consideraciones de certidumbre / incertidumbre en cuanto a software / sistema de comportamiento	1	8	
Prototipos	9		
Conceptos básicos de la especificación de los requerimientos formales	1	2	3
Especificación de requerimientos	9		
Validación de requerimientos	9		
Localización de requerimientos		9	

A continuación se citan algunos ejemplos que ilustran la aplicación de los contenidos de los distintos niveles propuestos por ACM [1].

En referencia al contenido de conocimiento del Nivel 1 “Describir los cambios fundamentales de técnicas utilizadas para la elicitación de requerimientos”, en la Tabla 4 se mencionan las mismas y los problemas asociados identificados en la implementación de los desarrollos propuestos

**Tabla 4.** Técnicas utilizadas y problemas identificados.

<b>Técnica</b>	<b>Problemas</b>
Entrevista o cuestionario	En la identificación de las preguntas claves para el caso de estudio
JAD (Desarrollo conjunto de aplicaciones)	En la asociación de las etapas de la técnica con el caso de estudio
Brainstorming	En la asociación de las etapas de la técnica con el caso de estudio
Casos de uso	En la comprensión para construir un diagrama que represente al caso de estudio

En referencia al contenido de conocimiento del Nivel 1 “Interpretar un modelo determinando los requisitos para un sistema de software sencillo”, los desarrollos abordados por el equipo, se plasmaron en un modelo compuesto por casos de uso, diagramas de secuencias y sus conversaciones y diagrama de clases.

Respecto al contenido de conocimiento del Nivel 2 “Llevar a cabo un examen de una serie de requisitos de software para determinar la calidad de los mismos con respecto a las características”, en los trabajos se aplicaron. Entre otros contenidos de conocimiento llevados a la práctica y que responden a este nivel se menciona: la enumeración de los componentes clave de un caso de uso o una descripción similar de algún comportamiento que se requiere para un sistema y su discusión en cuanto a su importancia en el proceso de Ingeniería de Requerimientos, la enumeración de los componentes fundamentales de un diagrama de clases o la descripción similar de los datos que requiere un sistema.

Como contenido de aplicación se determinaron las necesidades funcionales y no funcionales en una especificación de requisitos para un sistema de software.

Siguiendo lo expuesto por ACM [1], respecto a los resultados de aprendizaje, en las Tablas 5 a 7 se explicitan aquellos cumplidos según los distintos núcleos de contenidos.

**Tabla 5.** Resultados de aprendizaje del Núcleo de contenido 1

<b>Resultados de aprendizaje del Núcleo de contenido 1</b>	<b>SI</b>	<b>NO</b>
Describir los problemas fundamentales y las técnicas comunes utilizadas para relevar requerimientos. [Conocimiento]	5	4
Interpretar un modelo determinado los requisitos para un sistema de software sencillo. [Conocimiento]	8	1

**Tabla 6.** Resultados de aprendizaje del Núcleo de contenido 2

<b>Resultados de aprendizaje del Núcleo de contenido 2</b>	<b>SI</b>	<b>NO</b>	<b>NC</b>
Examinar una serie de requisitos de software para determinar su calidad con respecto a las características definidas como “buenos requisitos” [Aplicación]	3	2	4
Enumerar los componentes clave de un caso de uso o una descripción similar de algún comportamiento que se requiere para un sistema y discutir su papel en el proceso de IR [Conocimiento]	5	4	
Enumerar los componentes fundamentales de un diagrama de clases o la descripción similar de los datos que requiere un sistema para manejar [Conocimiento]	9		
Determinar las necesidades funcionales y no funcionales en una especificación dada requisitos para un sistema de software [Aplicación]	9		

**Tabla 7.** Resultados de aprendizaje del Núcleo de contenido 3

<b>Resultados de aprendizaje del Núcleo de Nivel electivas</b>	<b>SI</b>	<b>NO</b>
Aplicar los elementos clave y los métodos comunes para la obtención y el análisis para producir un conjunto de requisitos de un sistema de software de tamaño mediano. [Aplicación]	9	
Usar un método común, no formal para modelar y explicitar (en la forma de documento de especificación de requisitos, aquellos necesarios para un sistema de software de tamaño medio [Aplicación]	9	
Traducir al lenguaje natural, una especificación de requisitos de software (por ejemplo, un contrato componente de software), escrito en un lenguaje de especificación formal. [Aplicación]	9	
Crear un prototipo de un sistema de software para mitigar el riesgo en los requisitos. [Aplicación]	5	4

En relación a los resultados de aprendizaje del Núcleo de Nivel electivas se mencionan como ejemplos las siguientes experiencias. En el desarrollo de productos software se abordó el identificado como “Aplicar los elementos clave y los métodos comunes para la obtención y el análisis para producir un conjunto de requisitos de software para un sistema de software de tamaño mediano”.

El contenido de aplicación denominado “Usar un método común, no formal para modelar y explicitar -en la forma de documento de especificación de requisitos- aquellos necesarios para un sistema de software de tamaño medio” se plasmó en el empleo efectivo de una plantilla estandarizada. Permitió identificar los requerimientos funcionales y no funcionales, determinados por el caso de estudio presentado. Además, otra modalidad de especificar requerimientos son los casos de uso,

Se cumple el contenido de aplicación referido a “Traducir al lenguaje natural, una especificación de requisitos de software (por ejemplo, un contrato componente de software), escrito en un lenguaje de especificación formal”. Respecto al referido como “Crear un prototipo de un sistema de software para mitigar el riesgo en los requisitos”, es decir el riesgo que puede ocurrir en la aplicación cuando se vulnera la

seguridad, entre algunos inconvenientes que se pueden detectar se mencionan: infringir el código fuente, modificar o visualizar los datos almacenados en la base de datos. A fin de mitigarlo se consideraron algunas cuestiones relacionadas con la confidencialidad (encriptación de datos sensibles). En los distintos prototipos desarrollados, con la finalidad de validar los requerimientos del sistema se revisó la documentación confeccionada y se generaron las modificaciones en los casos correspondientes.

El relevamiento realizado en el equipo de trabajo, referente a la utilización de los ítems de los núcleos definidos por ACM [1] permitió analizar el conocimiento y aplicación de los mismos. En todos los desarrollos originados por el equipo, se acordó en el cumplimiento en el Nivel 1, indudablemente es el fundamento del análisis de requerimientos que debe estar presente. En referencia al núcleo Nivel 2, se coincide en un porcentaje del 33% para el ítem 1, un porcentaje del 56% para el ítem 2 y se presenta un acuerdo en los ítems 3 y 4.

Los resultados de aprendizaje para el Núcleo de Nivel Electivo, concuerdan en los tres primeros ítems, mientras que presenta una diferencia del 84% en el último.

## 4 Conclusiones

La revisión de los contenidos de los núcleos propuestos por ACM [1] y la evaluación destinada a los miembros que forman el grupo de investigación orientado a la I+D en referencia a la aplicación de técnicas de Ingeniería de Requerimientos, permitió,

- autoevaluar el grado actual de conocimientos y aplicación de los estándares fijados por ACM [1] hacia el interior del grupo de I+D.
- determinar su aplicación en la resolución de problemas del mundo real desde una perspectiva ingenieril, y consensuar o mediar entre los stakeholders, en este caso docentes y recursos humanos en formación en investigación.

Además, facilita a los participantes cumplir los objetivos propuestos, transfiriendo el conocimiento generado en el ámbito universitario al medio económico-productivo, específicamente de la Industria del Software. Esto también deriva en profesionales con formación universitaria acorde a los estándares fijados por organizaciones internacionales. Por otra parte, se continuará con la paulatina incorporación de los faltantes en las metodologías empleadas en futuros desarrollos con miras a alcanzar los niveles de calidad exigidos.

Se considera viable acordar aquellos contenidos del análisis de requerimientos, proponiendo una coherencia entre los analistas y desarrolladores atendiendo al tipo y plataforma de aplicación o sistema a desarrollar. Además esta actividad se plasma en el ámbito de la docencia entre articulaciones de asignaturas y el grupo de I+D.

Se abordó un estudio dialéctico, que refleja y se mueve en una constante espiral desde la teoría a la empiria, desde la investigación aplicada a la docencia en Educación Superior y su transferencia en conocimiento y modelos de representación de problemas, a fin de consolidar recursos humanos de grado y postgrado en temas especificados por el núcleo Ingeniería de Requerimientos, definido por ACM. Se espera que los resultados de este estudio que, serán puestos a consideración de las asignaturas que incorporan dichos contenidos en el Plan de Estudios actual, aporten

insumos que permitan una actualización constante y en pos de la calidad de la currícula de la carrera Licenciatura en Sistemas de Información o en posibles ofertas de formación de postgrado en la región del Nordeste Argentino de influencia preponderante de la UNNE.

## Referencias

1. ACM. Association for Computing Machinery. Computer Science Curricula 2013. IEEE-Computer Society, <http://ai.stanford.edu/users/sahami/CS2013/strawman-draft/cs2013-strawman.pdf>
2. Alderete, R., Escalante, E., Mariño S., Godoy, M.: Implementación del módulo “Verbos” del Prototipo educativo “Enseñanza para la lengua del 5º grado”. Anales del VI Congreso Argentino de Tecnologías en Educación y Educación en Tecnología. TE&ET’11 (2011).
3. ArgoUML, <http://argouml.tigris.org/>
4. Barrios, W., Godoy, M., Fernandez, M., Mariño S.: Sistema de apoyo a la gestión de salud pública rural: Etapas preliminares del proyecto. Simposio de Sociedad de la Información. 40º Jornadas Argentinas de Informática. Córdoba. Argentina (2011).
5. Barrios, W. Godoy Guglielmone, M., Fernandez, M., Mariño, S., Ferreira, F., Zarrabeitia, C.: SCRUM: Experiencia de Aplicación en una Empresa de Desarrollo de Software del NEA. WIS. Workshop de Ingeniería del Software. Congreso Argentino de Ciencias de la Computación. Universidad Nacional de La Plata. La Plata. Argentina (2011).
6. BOUML, <http://bouml.free.fr/>
7. Bulloni, D., Mariño, S., Godoy, M.: Prototipo de sistema de gestión y simulación de materiales. Un caso de estudio. Anales II ERABIO - XXIII ENDIO - XXI EPIO. ISBN 978-987-24257-1-2. 1085-1102pp. 18 págs. Tandil. Argentina, 15 al 17 de Septiembre de 2010. (2010).
8. IEEE STD-830. Recommended Practice for Software Requirements Specifications. Software Engineering Standards Committee of the IEEE Computer Society. (1998).
9. Jacobson, I., Rumbaugh, J., Booch, G.: El Proceso Unificado de Desarrollo del Software, Pearson Addison-Wesley (2000).
10. Loucopoulos, P., Karakostas V.: System Requirements Engineering. McGraw-Hill (1995).
11. Mariño S., Godoy, M., López, C., Matoso, A., Barreto, H., Escalante, J.: Generación de artefactos de software con tecnologías FLOOS y su aporte para la innovación y gestión de pequeñas empresas. En memorias del Sexto Encuentro en Línea de Educación Cultura y Software Libres. [En línea] México. Instituto de Investigaciones Económicas, Proyecto de investigación Psicoeducativa y Biné: La Comunidad Académica en Línea, <http://edusol.info/e2010/>. (2010).
12. Mariño, S., Godoy M., Lezcano, J., Zacarías, G.: 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. 21 y 22 de mayo de 2010. (2010).
13. Mariño S., Godoy, M., Esquivel, L., Paz, R., Torres, J., Escalante, J.: Las Tecnologías de la Información en la gestión de contenidos. Algunas soluciones transferidas al medio desde la UNNE”. Simposio de Sociedad de la Información. 40º Jornadas Argentinas de Informática. Córdoba. Argentina (2011).
14. Mariño S., Godoy, M., Acosta, J., Roa, D., Mendiburu, A. Algunas experiencias de vinculación Universidad – Sistema de Educación, mediante la generación de soluciones informáticas variadas. Anales VI Congreso Argentino de Tecnologías en Educación y Educación en Tecnología. TE&ET’11. (2011).

15. Mariño, S., Godoy, M., Escalante, J., Acosta, J., Roa, D., Sánchez, K., Lezcano, J., Zacarías, G., Bulloni, D., Schaeffer, M.: Construcción de software educativo. Síntesis de algunas experiencias 2009-2010. Hologramatica, Año VII, 14(4): 55-76. ISSN 1668-5024. (2011).
16. Pressman, R.: Ingeniería del Software: Un enfoque práctico. Sexta edición, México DF, Editorial McGraw Hill (2006).
17. Primorac, C., Mariño, S., Godoy, M.: TIC y Sociedad: Especificación de Requisitos para Apoyar la Gestión de Información. Revista Digital Sociedad de la Información. Monográfico TIC. N° 33. Enero 2012.
18. Rational Unified Process, Best Practices for Software, Development Teams
19. Sommerville, I.: Ingeniería del Software. Séptima edición, México DF, Editorial Pearson, (2005).
20. START UML, <http://staruml.sourceforge.net/en/>

## **Agradecimientos**

Se agradece la colaboración de Romina Alderete, Walter Barrios, Jaquelina E. Escalante, Carlos Primorac quienes contribuyeron a la realización del presente trabajo.