

LA ELICITACIÓN DE REQUISITOS EN EL CONTEXTO DE UN PROYECTO SOFTWARE

Markus Manies, Uolevis Nikual
Lappeenranta University of Technology, Finland
tite-toimisto@lut.fi

(Tipo de artículo: **REFLEXIÓN**. Recibido el 25/07/2011. Aprobado el 01/11/2011)

RESUMEN

La Elicitación de Requisitos –ER– es la piedra angular en el desarrollo de proyectos software y tiene un impacto muy alto en el diseño y en las demás fases del ciclo de vida del producto. Si se realiza apropiadamente, puede ayudar a reducir los cambios y las correcciones en los requisitos. Además, la calidad de la elicitación determina la exactitud de la retroalimentación al cliente acerca de la integridad y validez de los requisitos. Debido a que esta fase es crítica y de alto impacto en el proyecto, es muy importante que la labor de elicitar se realice lo más cercano posible a la “perfección”. Teniendo en cuenta las diferentes características de los proyectos software, en este trabajo se proponen algunas reglas generales para llevar a cabo la RE con base en la discusión y en la explicación de los procesos relacionados y métodos aplicados en los diferentes tipos de proyectos software.

Palabras clave

Ingeniería de Requisitos, Elicitación de Requisitos, Requisitos, Desarrollo de Software.

REQUIREMENTS ELICITATION IN THE CONTEXT OF A SOFTWARE PROJECT

ABSTRACT

Requirements elicitation –RE– is the cornerstone in the development of software projects and has a very high impact on the design and other phases of product's life cycle. If done properly, it can help to reduce changes and corrections in the requirements. Additionally, the quality of the elicitation determines the accuracy of the feedback given to the customer about the integrity and validity of the requirements. Because this phase is critical and has a profound impact on the project, it is very important that the elicit work can be made as perfect as possible. Considering the different features of software projects, this paper proposes some general rules for performing RE based on the discussion and the explanation of the related processes and methods applied in different types of software projects.

Keywords

Requirements Engineering, Requirements Elicitation, Requirements, Software Development.

L'ÉLICITATION DES EXIGENCES DANS LE CONTEXTE D'UN PROJET LOGICIEL

RÉSUMÉ

L'Élicitation des exigences –RE– (selon ses sigles en anglais) est la pierre angulaire dans le développement des projets logiciels, en ayant un impact très fort sur la conception et sur les autres phases du cycle de vie du produit. Si la RE est réalisée d'une manière appropriée, elle peut aider à réduire les changements et corrections sur les exigences. D'ailleurs, la qualité de l'élicitation détermine l'exactitude du feedback au client sur l'intégrité et validité des exigences. Cette phase est critique et de fort impact dans le projet, à cause de cette raison il est très important que la tâche d'éliciter soit réalisée le « mieux possible ». En considérant les différentes caractéristiques des projets logiciels, dans ce travail on propose quelques règles générales pour accomplir la RE selon la discussion et l'explication des processus liés et des méthodes appliqués dans les différents types des projets logiciels.

Mots-clés

Génie des exigences, Élicitation des exigences, Exigences, Développement des logiciels

1. INTRODUCCIÓN

Es una realidad que entre el 40% y el 60% de los errores y defectos del software son el resultado de una pobre gestión y definición de requisitos. En palabras simples, esto significa que aproximadamente la mitad de los problemas encontrados se podrían haber evitado, simplemente con dejar claro, desde el principio, lo que el cliente espera del proyecto respectivo.

En ayudar en esta situación, se propone la Ingeniería de Requisitos –IR. La IR es la ciencia y la disciplina que se ocupa de establecer y documentar los requisitos del software [1]. La versión 1.2 del Capability Maturity Model Integration-Development –CMMI– divide los requisitos del software en dos áreas de proceso: la primera es el Desarrollo de Requisitos, que incluye la elicitación, la definición, el análisis, la especificación y la validación; la segunda es la Gestión de Requisitos, que implica la gestión de requisitos que se han desarrollado, incluyendo el control de cambios y la verificación.

La Elicitación de Requisitos –ER– se considera como la primera etapa en el proceso de abstraer una comprensión del problema que se quiere resolver con el producto software. Se trata, esencialmente, de una actividad humana donde se identifican las partes interesadas y se establecen las relaciones entre el comprador, el cliente, los usuarios y el equipo de desarrollado. El término Elicitación lo utiliza la comunidad para resaltar el hecho de que los buenos requisitos no sólo se obtienen desde los clientes, como se indica cuando se utiliza la frase “recopilación de requisitos”. Christel & Kang [2] identifican una serie de problemas que ayudan a comprender por qué es difícil la Elicitación de Requisitos:

1. Problemas de alcance. Las fronteras del sistema están mal definidas o los clientes/usuarios definen detalles técnicos innecesarios que pueden confundir, más que aclarar, los objetivos generales del sistema.
2. Problemas de comprensión. Los clientes/usuarios no están completamente seguros de lo que necesitan, tienen una comprensión deficiente de las capacidades y limitaciones de su entorno informático, no tienen una plena comprensión del dominio del problema, tienen dificultades para comunicar sus necesidades al ingeniero de software, omiten información que consideran “evidente”, definen requisitos que entran en conflicto con las necesidades de otros clientes/usuarios, o definen requisitos que son ambiguos o improbables.
3. Problemas de volatilidad. Los requisitos cambian con el tiempo.

Para ayudar a superar estos problemas, los ingenieros de software deben ejecutar de forma organizada la captura de requisitos.

La ER es el componente básico sobre el que se construye un proyecto software y tiene un impacto muy alto en el diseño y las fases posteriores del ciclo de vida del producto. Si se realiza apropiadamente, puede ayudar a reducir los cambios y las correcciones en los requisitos. Además, la calidad de la elicitación determina la exactitud de la retroalimentación al cliente acerca de la integridad y validez de los requisitos [3]. Debido a que esta fase es crítica y de alto impacto en el proyecto, es muy importante que la labor de elicitar se realice lo más cercano posible a la perfección.

Las prácticas de ER incluyen entrevistas, cuestionarios, observación a la labor del usuario, talleres, lluvia de ideas, casos de uso, juegos de rol y la creación de prototipos; aunque existen muchas otras características diferentes en los proyectos reales. Con base en la experiencia de algunos investigadores en proyectos de la vida real, a continuación se resumen los enfoques empleados para elicitar requisitos en diversas circunstancias, de este modo se reducen las dificultades de esta ciencia y se incrementa la eficiencia de la elicitación.

2. PRÁCTICAS DE LA IR

Los proyectos informáticos pueden estructurarse para actualizar un sistema antiguo o para iniciar un nuevo sistema. El proceso de iniciar un nuevo sistema puede dividirse en las siguientes cuatro categorías [4], dependiendo del grado de claridad que tengan las partes interesadas de las necesidades del mismo: 1) tanto los desarrolladores como los clientes tienen claros los requisitos de proyecto, 2) los desarrolladores no los tienen claros pero el cliente sí, 3) ni los desarrolladores ni el usuario los tienen claros y 4) los desarrolladores los tienen claros pero del lado del cliente no.

2.1 Proyectos para actualizar sistemas existentes

Con el desarrollo de la informatización, cada vez más empresas tienen su propio sistema de gestión de la información. Sus usuarios plantearán nuevos requisitos en los proyectos debido a nuevas funciones añadidas, a cambios en la lógica de negocio o a deficiencias en el sistema antiguo, lo que hace que cada vez sea más difícil satisfacer las necesidades de las empresas. En estos sistemas, los pasos para elicitar requisito son los siguientes:

1. Familiarización con el sistema antiguo. La razón por la cual el sistema existe hasta el momento es porque puede satisfacer la mayoría de las necesidades de la empresa, especialmente en el core del negocio, o porque se le han hecho actualizaciones acumuladas. Por lo tanto, la mayoría de los requisitos del software se pueden capturar si se logra una buena familiarización con ese sistema. Además, es recomendable la familiarización con el conocimiento de la industria en campos relacionados, con el objetivo de reducir más tarde las barreras en la comunicación con el usuario. Si existen documentos pertinentes del sistema antiguo, también se pueden aprovechar.

2. Entrevistas. Luego del paso anterior se obtiene un buen conocimiento de la mayoría de los requisitos del sistema y se logra la familiarización con su contexto; a continuación, se procede a entrevistar a los usuarios. A través de entrevistas con los diferentes usuarios se conocen los requisitos para cambiar o incrementar la demanda. En términos generales, cuando los usuarios están muy dispuestos a hablar con los ingenieros y se explican de forma clara y detallada acerca de lo que requieren, esas piezas que describen son las que necesitan ser modificadas o añadidas y son exactamente lo que el viejo sistema no podía satisfacer de sus necesidades, por lo que han estado preocupados durante mucho tiempo.

2.2 Proyectos para iniciar un nuevo sistema

Ambas partes tiene claros los requisitos

Para estos proyectos, comúnmente se utiliza el método de encuesta con el objetivo de lograr una mejor comprensión, ya que tanto para los desarrolladores como para el cliente los requisitos del proyecto son claros y existen pocas necesidades o problemas que necesiten una comunicación más amplia de las partes. El "método de encuesta" se refiere a una forma de elicitar requisitos, en la que el desarrollador utiliza un cuestionario que envía a los usuarios para lograr una comprensión completa del proyecto con base en sus demandas individuales y en las demandas –o problemas– necesarios para definirlos con mayor precisión.

El cuestionario se utiliza ampliamente como herramienta para la elicitación de requisitos tal y como se utiliza en la ciencia para el análisis estadístico. Los cuestionarios se utilizan cuando:

- El número de personas es grande.
- Se necesitan respuestas a problemas específicas bien definidos.
- Se quiere un resultado específico.

Durante la preparación de cuestionarios hay que tener en cuenta:

1. Mantener el cuestionario lo más pequeño posible. En lugar de utilizar un cuestionario grande, es mejor aplicar varios pequeños. En el caso de los cuestionarios grandes, por lo general, el usuario tiende a cansarse después de contestar las primeras 15-20 preguntas y no será muy objetivo al responder el resto. Como regla común un cuestionario no deberá contener más de 10-15 preguntas.
2. Estimar el tiempo necesario para responder las preguntas y un ambiente propicio que facilite la objetividad del cuestionario.
3. Estar seguro que las preguntas están en un contexto libre de ambigüedades. Para asegurarlo, aplicar el prototipo a alguien cercano a la

comunidad de los entrevistados y confirmar su comprensión de las preguntas estructuradas.

4. Antes de formular preguntas, estar seguro de que se necesitan las respuestas. En un contexto específico, las respuestas a muchas preguntas se convierten en información sin sentido.
5. Listar todas las posibles preguntas separadamente. Una vez que todos los requisitos y preguntas están listos, crear un plano cartesiano, con los requisitos en el eje X y las preguntas sobre el eje Y; para cada pregunta, identificar cuáles de los requisitos se están cumpliendo. Al final de este ejercicio se descartan las preguntas que no estén asociadas a uno de los requisitos.

Este método tiene la ventaja de ser sencillo y claro, lo que podría acortar el tiempo, reducir los costos y mejorar la eficiencia de la captura de requisitos.

Los clientes conocen los requisitos pero los desarrolladores no

Para estos proyectos, la cuestión aparentemente es muy fácil. Parece que todo lo que se necesita es llevar a cabo algunas reuniones con los clientes para discutir sus necesidades, a fin de lograr el sistema que las satisfaga. La práctica demuestra que esto es inviable, ya que los desarrolladores no conocen los requisitos, es decir, que no están familiarizados con el ámbito del proyecto. Como resultado, no saben cómo guiar a los clientes para discutir este problema y éstos se limitan sólo a mencionar las cuestiones generales sin entrar en detalles, debido a que creen que el problema no existe en absoluto y que todo el mundo debería conocerlo. Lo que es peor, los desarrolladores no pueden hacerles preguntas a los clientes de acuerdo con sus descripciones, debido a que carecen de conocimientos en la materia. Después de la discusión, los clientes piensan que han explicado claramente los problemas, mientras que los desarrolladores no han comprendido absolutamente nada.

Cada sistema de información en realidad es una caja negra: por un extremo ingresan los datos y por el otro surge la información. Si es posible conseguir ambas cosas, significa que se tiene una idea aproximada del sistema. Para estos sistemas se recomienda:

1. Utilizar métodos de compilación para los datos de entrada y de salida. En este proceso, se recogen todas las formas que contienen las necesidades de los usuarios, todo el material impreso como las formas de reportes necesarios y se recuperan para análisis y compilación. En este proceso se recoge la mayor parte de necesidades de los usuarios con el objetivo de conocer el contexto. Este proceso también puede funcionar en empresas sin un sistema de informatización. Sólo porque muchas cosas se adquieren impresas, la mayoría de estas necesidades se puede conocer para reunir, analizar y clasificar la información. Por lo tanto, también es un proceso en el que los desarrolladores se

familiarizan con el campo y adquieren las bases para futuras conversaciones y para la comunicación con los usuarios.

2. Reuniones de grupo. Después de llevar a cabo el primer paso, se tiene una comprensión bastante profunda del sistema. Las reuniones de grupo se pueden utilizar comúnmente en los siguientes casos:

- El conocimiento se distribuye por igual entre un grupo reducido de personas.
- Incapacidad para satisfacer todas las inquietudes de las partes interesadas individualmente.
- Se ha llevado a cabo una serie de entrevistas y el equipo necesita tener a todos los participantes en una misma sala.

En la reunión de grupo, cada participante puede expresar sus pensamientos. Las respuestas de un grupo son mejores que las respuestas de un individuo, para estos casos. Las reuniones de grupo también ayudan a reducir el número de conflictos en los requisitos, así como a mantener una tabulación de los deseos más locos.

Estas reuniones se llevan a cabo en el espacio del cliente. Si las reuniones de grupo se gestionan mal, tienden a causar los siguientes problemas: 1) un pequeño número de participantes toma el control de la discusión, o 2) algunos de los participantes no se entusiasman a hablar. Para evitar estas situaciones, se necesita un analista para promover la discusión. Se debe animar a los participantes a que se entusiasmen en la discusión: al principio, pedirles que propongan algunos problemas cerrados y luego poco a poco convertirlos en cuestiones abiertas. También se deben tener en cuenta ciertas reglas básicas como el uso de preguntas abiertas, retomar temas para confirmar su comprensión, definir una agenda clara, utilizar personas que tomen atenta nota, entre otras.

Los desarrolladores conocen los requisitos pero los clientes no

Lo primero que se debe hacer es dejar que el cliente reconozca lo que realmente quiere. Los métodos de "prototipos" y las "reuniones de discusión" podrían utilizarse aquí. El proceso es:

1. Los prototipos de interfaces se basan en la comprensión que tienen los desarrolladores de los requisitos del cliente; entonces, modelan una interfaz de la aplicación y la utilizan para comunicarse con los clientes. Con el método de prototipos ambos actores podrían alcanzar gradualmente la confirmación de los requisitos del proyecto. Los siguientes son los pasos generales de este método [5]:

Paso 1. Con su comprensión de los requerimientos y de las herramientas de diseño, los desarrolladores trazan prototipos de la aplicación.

Paso 2: Los desarrolladores presentan estos prototipos al cliente y los discuten con ellos, lo que ayuda a descubrir nuevos requisitos o a alcanzar un acuerdo.

Paso 3: De forma incremental, los desarrolladores analizan estos requisitos y adicionan nuevos prototipos y mejoran los existentes.

Paso 4: Al comunicarse en repetidas ocasiones sobre la base de estos prototipos, los desarrolladores finalmente logran la "especificación de requisitos de usuario".

La incompreensión de los requisitos del proyecto, por parte del cliente, puede incrementar las dificultades y los riesgos del proceso de captura. El "método de prototipos" puede acelerar el descubrimiento de nuevos requisitos y su aceptación por las partes y, al mismo tiempo, reducir los posibles riesgos.

2. Reuniones de discusión. Después de alcanzar el acuerdo básico sobre el problema de "qué hacer" en el proceso anterior, las partes llevan a cabo reuniones para comprender el sistema de reglas de negocio, que consiste en "cómo implementar el modelo prototipado". Este método de "reuniones de discusión" también puede ayudar a determinar los requisitos del proyecto.

Ambas partes desconocen los requisitos

Esta condición presenta grandes dificultades para la captura de requisitos y es el mayor riesgo en el desarrollo del sistema. En esta situación se pueden utilizar los métodos de "investigación" y de "escenarios".

El "método de investigación" colecta una amplia gama de requisitos a través de:

- Investigar el trabajo de otras empresas en el mismo contexto.
- Observar sistemas similares.
- Revisar los documentos del sistema actual o el anterior.

Durante la fase de investigación se requiere información como:

1. ¿Por qué necesita el cliente un sistema así? ¿Qué tipo de ayuda le proporciona al cliente el nuevo sistema? ¿Qué efecto tendrá el nuevo sistema sobre la eficiencia en el trabajo?
2. ¿Qué es lo que desea el cliente que este sistema haga o qué funciones implementa este sistema?
3. ¿Qué condiciones de manipulación se necesitan de las funciones de este sistema?

Los escenarios también ayudan a elicitar requisitos en este tipo de proyectos [6]. Con la Elicitación de Requisitos basada en escenarios, es posible mapear el planteamiento del problema del sistema en una especificación que se representa como un conjunto de actores y de casos de uso. Los actores representan las entidades externas que interactúan con el sistema. El primer paso de la Elicitación de Requisitos es la identificación de los actores. Una vez identificados los actores y mediante la identificación de escenarios, se determina la funcionalidad que cada uno necesita abordar.

Un escenario es una descripción del uso del sistema en términos de una serie de interacciones entre el sistema y el usuario. El trabajo con los clientes es necesario para lograr un conjunto completo de escenarios, que se describen en lenguaje natural utilizando su terminología. Ese conjunto completo de escenarios describe todo lo que se pretende que el sistema haga. A continuación, los desarrolladores formalizan los escenarios en casos de uso. Un escenario es una instancia de un caso de uso. Un caso de uso especifica todos los escenarios posibles para una determinada pieza de funcionalidad. Un caso de uso lo inicializa un actor.

3. RECOMENDACIONES

Estos métodos no son mutuamente excluyentes y pueden aplicarse de forma separada o integrada, de acuerdo con las características del proyecto. Existen otros métodos, como las llamadas telefónicas, el correo electrónico y el MSN, que también se pueden adoptar cuando los requisitos del sistema se han capturado y quedan algunos detalles sin resolver. Sin importar qué método se aplique, existen algunas reglas que deben tenerse en cuenta. Sommerville y Sawyer [7] sugieren un conjunto de directrices detalladas para la elicitación de requisitos:

- Evaluar la viabilidad comercial y técnica para el sistema propuesto.
- Identificar las personas que ayudarán a especificar los requisitos y a comprender sus prejuicios organizacionales.
- Definir el entorno técnico –por ejemplo, la arquitectura computacional, el sistema operativo– en el que funcionará el sistema o producto.
- Identificar las "restricciones de dominio" –es decir, las características del entorno empresarial

específico para el dominio de aplicación– que limitan la funcionalidad o el rendimiento del sistema o producto que se desarrolla.

- Definir uno o más métodos de Elicitación de Requisitos –por ejemplo, entrevistas, grupos focales, reuniones de grupo.
- Solicitar la participación de muchas personas para que se definan los requisitos desde diferentes puntos de vista; asegurarse de identificar la base lógica de cada requisito que se registra.
- Identificar requisitos ambiguos como candidatos a prototipar.
- Crear escenarios de uso para ayudarles a los clientes/usuarios a identificar mejor los requisitos.

4. CONCLUSIONES

La clave para lograr el éxito en un proyecto software es la Elicitación de Requisitos. Ante la variedad de métodos de elicitación y los diferentes niveles de madurez de los requisitos de cada proyecto, se debe adoptar una estrategia correcta para llevarla a cabo. En este trabajo se describen algunos métodos para aplicar, de acuerdo con el tipo de proyecto que se inicia. Seleccionar un método correcto puede acortar el tiempo, reducir los costos y mejorar la eficiencia de la Elicitación de Requisitos.

REFERENCIAS

- [1] R. H. Thayer. "Software Requirements Engineering". Wiley-IEEE Computer Society Press, 1997.
- [2] M. G. Christel & K C. Kang. "Issues in Requirements Elicitation". Technical Report CMU/SEI-92-TR-012, ESC-TR-92-012. Software Engineering Institute, Carnegie Mellon University, 1992.
- [3] S. Robertson & J. Robertson. "Mastering the Requirements Process". Addison-Wesley, 1999.
- [4] K. E. Wiegers. "In Search of Excellent Requirements Process Impact". [Mar. 2011].
- [5] D. Zowghi & C. Coulin. "Requirements Elicitation: A Survey of Techniques, Approaches, and Tools". In A. Aurum & C. Wohlin (Eds.) "Engineering and Managing Software Requirements" pp. 19-46. Springer-Verlag, 2005.
- [6] L. Feng et al. "A Scenario-Based Collaborative Requirements Elicitation Approach for Enterprise Information Systems". *Acta electronica SINICA*. Vol. 37, No. 4, pp. 51-57, 2009.
- [7] I. Sommerville & P. Sawyer. "Requirements Engineering: A Good Practice Guide". John Wiley & Sons, 1997.