

EL DESARROLLO DE PROYECTOS DE SOFTWARE Y LA IMPORTANCIA DE LA INGENIERÍA DE REQUERIMIENTOS

Carmen Janeth Padilla Vedia

Dpto. Informática y Sistemas - Facultad de Ciencias y Tecnología - Universidad Autónoma Juan Misael Saracho

Correo electrónico: padillac555@gmail.com

RESUMEN

Al realizar este artículo se trata de reflejar una panorámica acerca de los conceptos y características de la Ingeniería de Requerimientos (IR), buscando resaltar su relevancia dentro del ciclo de desarrollo de proyectos de software, conocer las diferentes alternativas o técnicas que existen para identificarlos, analizarlos, documentarlos, así como mostrar la importancia que tienen herramientas automatizadas dentro de este proceso de administración de requerimientos.

PALABRAS CLAVE:

Requerimientos, ingeniería de requerimientos, herramientas, técnicas, wizard, stakeholders, brainstorming, elicitación

PROBLEMA DE INVESTIGACIÓN

Todo proceso de desarrollo de software implica la identificación o captura de requerimientos, los mismos que se encuentran de forma explícita o implícita en el ambiente del proyecto involucrado, situación que conlleva a trabajar desde la primer etapa, fase o como se lo denomine en la metodología seleccionada.

Esta situación nos lleva a todos los involucrados con la Ingeniería de Requerimientos a pensar y decidir en función a las particularidades de cada proyecto ¿Cómo enfrentamos el proceso inicial de desarrollo de software ?.

La interrogante expresada líneas atrás debe permitir reflejar que haremos para captar los requerimientos, de ahí que presentamos las técnicas, herramientas manuales y automatizadas que permitan realizar esta actividad de una forma metodológica garantizando calidad en el proceso.

I. INTRODUCCIÓN

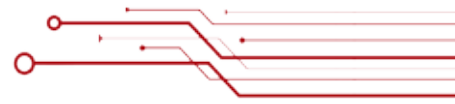
A través de los años se ha podido constatar que los requerimientos o requisitos son la pieza fundamental en un proyecto de desarrollo de software, ya que marcan el punto de partida para actividades como la planeación, básicamente en lo que se refiere a las estimaciones de tiempos y costos, así como la definición de recursos necesarios y la elaboración de cronogramas que será uno de los principales mecanismos de control con los que se contará durante la etapa de desarrollo. Además la especificación de requerimientos es la base que permite verificar si se alcanzaron o no los objetivos establecidos en el proyecto ya que estos son un reflejo detallado de las necesidades de los clientes o usuarios del sistema y es contra lo que se va a estar verificando si se están cumpliendo las metas trazadas.

Un número creciente de herramientas automatizadas han surgido para ayudar a definir y aplicar un proceso de desarrollo de software efectivo.

Hoy en día la economía global depende más de sistemas automatizados que en épocas pasadas; esto ha llevado a los equipos de desarrollo a enfrentarse con una nueva década de procesos y estándares de calidad.

Sin embargo, ¿cómo explicamos la alta incidencia de fallos en los proyectos de software? ¿Por qué existen tantos proyectos de software víctimas de retrasos, presupuestos sobregirados y con problemas de calidad? ¿Cómo podemos tener una producción o una economía de calidad, cuando nuestras actividades diarias dependen de la calidad del sistema?

Tal vez suene ilógico pero, a pesar de los avances



que ha dado la tecnología, aún existen procesos de producción informales, parciales y en algunos casos no confiables.

Es muy frecuente escuchar entre los conocedores del desarrollo de software (programas de computadoras), que un gran número de los proyectos de software fracasan por no realizar una adecuada definición, especificación, y administración de los requerimientos. Dentro de esa mala administración se pueden encontrar factores como la falta de participación del usuario, requerimientos incompletos y el mal manejo del cambio a los requerimientos.

La Ingeniería de Requerimientos (IR) cumple un papel primordial en el proceso de producción de software, ya que se enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; de esta manera, se pretende minimizar los problemas relacionados por la mala gestión de los requerimientos en el desarrollo de sistemas.

La Ingeniería de Requerimientos (IR) cumple un papel primordial en el proceso de producción de software, ya que se enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; de esta manera, se pretende minimizar los problemas relacionados por la mala gestión de los requerimientos en el desarrollo de sistemas.

Documentación de Requerimientos

Los documentos de la Ingeniería de Requerimientos son largos, está compuesto de un número grande de páginas que contienen detalles que pueden tener efectos profundos para el resto del sistema ya que esta etapa se convierte en el hilo conductor del proceso de desarrollo.

De hecho aquí surge un problema que las personas se encuentran con dificultades para leer y comprender un documento de esta magnitud, lo

que puede causar problemas en recordar todos los elementos, omitir situaciones relevantes y hasta cometer errores que se evidencian en etapas posteriores del proceso de desarrollo.

Razón por la cual las actividades que implica la Ingeniería de Requerimientos son varias y son aplicadas de manera continua y en orden como se presenta en el siguiente punto

2. REQUERIMIENTOS E INGENIERÍA DE REQUERIMIENTOS

Para poder llegar a un entendimiento más acabado de lo que la ingeniería de requerimiento es y qué representa, antes debemos entender qué son los requerimientos, cómo están definidos y para qué sirven. Existen varias definiciones, entre las que podemos citar:

2.1.¿Que son los Requerimientos?

- son expresiones de las necesidades de stakeholders para alcanzar una meta particular. [Nuseibeh]
- una condición o capacidad necesaria dada por un usuario con el objetivo de resolver un problema o alcanzar un objetivo [IEEE610:1990]
- Expresan las necesidades y restricciones atribuibles a un producto de software que contribuye a la solución de algún problema del mundo real [Kotonya:2000]
- Un requerimiento es simplemente una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste". (Sommerville, 2005: 108)

Los requerimientos pueden ser categorizados en varios niveles de abstracción, importancia, alcance, exactitud y detalle [COMP]. Por ejemplo:

- Requerimientos muy generales que expresan con términos amplios qué es lo que el sistema debería hacer.
- Requerimientos funcionales que definen partes de la funcionalidad del sistema.

- Requerimientos no funcionales que agregan restricciones al desarrollo del sistema
- Requerimientos de implementación que declaran cómo el sistema debe ser implementado.
- Requerimientos de performance que especifican una performance mínima aceptable para el sistema
- Requerimientos de usabilidad que especifican el tiempo máximo aceptable para demostrar el uso del sistema

Los requerimientos pueden ser clasificados como funcionales o de calidad. Los requerimientos funcionales describen qué es lo que el sistema debe hacer para dar soporte a las funciones y objetivos del usuario [Sommerville:1997]. Deben responder a las siguientes preguntas [DOE]:

- Cómo las entradas son transformadas en salidas?
- Quién inicia y recibe información específica?
- Qué información debe estar disponible para que cada función sea ejecutada?

Los requerimientos de calidad o no funcionales imponen restricciones de cómo los requerimientos funcionales deben ser implementados [Sommerville].

De esta manera una vez definidos los requerimientos, se intentará responder a las preguntas ¿qué es? y ¿qué abarca? la Ingeniería de Requerimientos

2.2. Ingeniería de Requerimientos

La ingeniería de requerimientos define procesos sistemáticos, que introducen una estrategia sólida para derivar una definición de productos de software, es por ello que su objetivo es establecer los requisitos del sistema a desarrollar y su gestión posterior.

Estos procesos sistemáticos establecen una solución de calidad a las necesidades reales del cliente, los límites bajo los cuales operan y se desarrollan. Los requisitos quedan plasmados en un documento, en

el cual se reflejan los acuerdos alcanzados entre clientes, usuarios y desarrolladores.

Los procesos de ingeniería de Requerimientos varían frecuentemente de una organización a otra, dependiendo de las tecnologías y métodos utilizados para el desarrollo, de la cultura organizacional y del área de aplicación.

Concretamente algunos autores definen a la Ingeniería de Requerimientos como sigue:

- La Ingeniería de Requerimientos es un proceso cíclico que involucra: la elicitación, especificación, validación y verificación de requerimientos [Hans van Vliet].
- La Ingeniería de Requerimientos se ocupa de la elicitación, análisis, especificación y validación de requerimientos de software [SWE-BOK:2004].
- 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”. (Pressman, 2006: 155)
- 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”. (Sommerville, 2005)

En otras palabras, la ingeniería de requerimientos involucra el descubrir cuáles son las metas, necesidades y expectativas de los stakeholders, ajustar las expectativas de los mismos y comunicarlas a los desarrolladores [Nuseibeh].

La ingeniería de requerimientos está formada por una serie de procesos bien diferenciados [Wiegers:1999].

2.2.1. Desarrollo de requerimientos

2.2.1.1. Elicitación



Esta fase representa el comienzo de cada ciclo. Elicitación o extracción es el nombre comúnmente dado a las actividades involucradas en el descubrimiento de los requerimientos del sistema. Aquí, los analistas de requerimientos deben trabajar junto al cliente para descubrir el problema que el sistema debe resolver, los diferentes servicios que el sistema debe prestar, las restricciones que se pueden presentar, etc.

Es importante, que la extracción sea efectiva, ya que la aceptación del sistema dependerá de cuán bien éste satisfaga las necesidades del cliente.

2.2.1.2 Análisis

En el proceso de análisis de requerimientos se trata, precisamente, de analizar la información recibida desde los usuarios, para distinguir las necesidades de tareas, los requerimientos funcionales, atributos de calidad, soluciones sugeridas, de información extraña [Wiegers:1999].

Se define a la etapa de análisis de requerimientos como la actividad de transformar requerimientos informales en requerimientos técnicos mediante el aseguramiento de que los mismos reflejan los atributos de calidad de los requerimientos y que expresan las necesidades de los clientes. El análisis es una actividad iterativa. Los pasos del proceso deberán ser repetidos una cierta cantidad de veces, existe una constante comunicación entre la consulta de los clientes, los usuarios finales y los desarrolladores [Avionics:1998].

Es el proceso de razonamiento acerca de los requerimientos que han sido elicitados; involucra actividades como el examen de requerimientos en busca de conflictos o inconsistencias, la combinación de requerimientos relacionados y la identificación de requerimientos perdidos [Tuffley:2005].

2.2.1.3 Especificación

No cabe ninguna duda de la importancia de esta etapa y de que la forma de especificar tiene mucho que ver con la calidad de la solución. Los analistas que se han esforzado en trabajar con especificaciones incompletas, inconsistentes o mal establecidas han experimentado la frustración y confusión que invariablemente se produce. Las consecuencias se

padecen en la calidad, oportunidad e integridad del software resultante.

Este proceso se encarga del grabado o el registro de los requerimientos en una o más formas, incluyendo el lenguaje natural y formal, representaciones simbólicas o gráficas [Tuffley:2005]. La especificación de los requerimientos es el paso en donde los resultados de la identificación de los requerimientos se “retratan” [Brackett:1990].

“La ambigüedad es la causa raíz de la mayoría de los problemas del entendimiento y documentación de los requerimientos.”

Como regla general, los requerimientos funcionales que describen qué es lo que el sistema y el software deben hacer deben estar separados de los requerimientos no funcionales que son los requerimientos de atributos de calidad especificados por el cliente, como son confiabilidad, seguridad y escalabilidad [Borland:2005].

Los requerimientos, tradicionalmente, se representan en una forma puramente textual. Sin embargo, incrementalmente se está utilizando técnicas como construcción de modelos y prototipos, que demandan una descripción más detallada de los requerimientos.

Dado que los usuarios muchas veces no son capaces de pensar en todas las situaciones posibles en las cuales el software puede ser utilizado, es tarea de los desarrolladores documentar los requerimientos desde un punto de vista que admita el testing. En este proceso se darán a conocer posibles situaciones que no se hubiesen tenido en cuenta la primera vez [Borland:2005].

2.2.1.4. Verificación

La validación es la etapa final de la IR. Su objetivo es, ratificar los requerimientos, es decir, verificar todos los requerimientos que aparecen en el documento especificado para asegurarse que representan una descripción, por lo menos, aceptable del sistema que se debe implementar. Esto implica verificar que los requerimientos sean consistentes y que estén completos.

Se puede apreciar que el proceso de ingeniería

de requerimientos es un conjunto estructurado de actividades, mediante las cuales se obtiene, se valida y se logra dar un mantenimiento adecuado al documento de especificación de requerimientos, que es el documento final, de carácter formal, que se obtiene de este proceso. Es necesario recalcar que no existe un proceso único que sea válido de aplicar en todas las organizaciones. Cada organización debe desarrollar su propio proceso de acuerdo al tipo de producto que se esté desarrollando, a la cultura organizacional, y al nivel de experiencia y habilidad de las personas involucradas en la ingeniería de requerimientos. Hay muchas maneras de organizar el proceso de ingeniería de requerimientos utilizando unas u otras herramientas, técnicas que no son explícitas por los diferentes autores que trabajan y proponen estas etapas para el desarrollo de requerimientos.

Este proceso puede implicar las siguientes actividades:

Ejecutar una evaluación: para asegurar que los requerimientos de calidad han sido alcanzado apropiadamente, se han descubierto las inconsistencias entre los requerimientos, se han identificado las redundancias, y el impacto de los requerimientos derivados.

Verificar la trazabilidad: verificar que todos los requerimientos se conectan con un requerimiento de más alto nivel, verificar que todos los requerimientos de alto nivel se conectan con un requerimiento formalizado e identificar requerimientos que no se encuentran anidados en otros, es decir que no poseen un “requerimiento padre”.

Documentar los hallazgos: creación de reportes que contenga los requerimientos junto a una definición más acabada.

Acuerdo: el documento formalizado de requerimientos será utilizado como un acuerdo entre el cliente y el desarrollador.

Establecer una línea base de los requerimientos: poner bajo administración de la configuración al documento de requerimientos.

2.2.2 Administración de requerimientos

No es suficiente para las organizaciones recolectar requerimientos desde múltiples stakeholders e incorporarlos independientemente dentro de un sistema.

Es necesario administrar estos de manera simultánea, además alguien, o algún equipo, debe ser responsable de administrar esos requerimientos a lo largo del ciclo de vida de manera tal de mantener la visibilidad y control del proceso de entrega del software [Borland:2005].

El acuerdo de los requerimientos es el puente que une el desarrollo de los requerimientos y la administración de los mismos. La administración de los requerimientos incluye todas las actividades que mantienen la integridad y exactitud de los requerimientos a medida que el proyecto progresa. Actividades tales como:

- Control de los cambios a los requerimientos que están sobre una línea base.
- Mantener los planes del proyecto actualizados de acuerdo con los requerimientos.
- Control de versiones tanto de requerimientos individuales como del documento de requerimientos.
- Manejar las relaciones entre requerimientos, y los links y dependencias entre los requerimientos individuales y otros entregables del proyecto.
- Monitorear el estado de los requerimientos sobre una línea base.

3 TÉCNICAS UTILIZADAS EN LA IR

Existen varias técnicas sin embargo se mencionaran algunas de las más importantes

3.1 Técnicas de elicitación.

El análisis de requerimientos siempre comienza con una comunicación entre dos o más partes. En el libro Ingeniería de Software de R. Pressman [Pressman], nos sugiere que un cliente tiene un problema al que puede encontrar una solución basada en computadora. El desarrollador responde



a la petición del cliente. La comunicación ha comenzado. Pero, el camino entre la comunicación y el entendimiento está lleno de baches.

Antes de mantener las reuniones con los clientes y usuarios e identificar los requerimientos es fundamental conocer el dominio del problema. Enfrentarse a un desarrollo sin conocer las características principales ni el vocabulario propio de su dominio suele provocar que el producto final no sea el esperado por clientes ni usuarios. Por otro lado, mantener reuniones con clientes y usuarios sin conocer las características de su actividad hará que probablemente no se entiendan sus necesidades y que su confianza inicial hacia el desarrollo se vea deteriorada enormemente.

Para conocer el dominio del problema se puede obtener información de fuentes externas al negocio del cliente: folletos, informes sobre el sector, publicaciones, consultas con expertos, etc. En el caso de que se trate de un dominio muy específico puede ser necesario recurrir a fuentes internas al propio negocio del cliente, en cuyo caso pueden utilizarse las técnicas de elicitación de requerimientos como el estudio de documentación, observación in situ, cuestionarios, etc.

En realidad una primera reunión entre el cliente y el analista servirá como un período corto de preguntas y respuestas, el cual, en adelante debe sustituirse por reuniones que busquen entender el problema del usuario.

Normalmente encontramos que los clientes y analistas se enfrascan en el proyecto de forma unilateral y no en equipo. Cada parte define su propio "territorio" y se comunica a través de una serie de notas, impresos formales, documentos y sesiones de preguntas y respuestas. Este enfoque no es muy efectivo, abundan los malentendidos, se pierde información importante y nunca se establece una relación de trabajo satisfactoria.

Con estos problemas presentes, se desarrollaron numerosas técnicas para tratar de superar este difícil momento, que es el inicio del proceso. Cada técnica puede aplicarse en una o más actividades de la ingeniería de requerimientos; en la práctica, la técnica más apropiada para cada actividad

dependerá del proyecto que esté desarrollándose.

3.1.1 Entrevistas

Las entrevistas son la técnica de elicitación más utilizada, y de hecho son prácticamente inevitables en cualquier desarrollo. En las entrevistas se pueden identificar claramente tres fases [Piattini]: preparación, realización y análisis, que se describen a continuación

Preparación de entrevistas

Las entrevistas no deben improvisarse, por lo que conviene realizar las siguientes tareas previas:

- Estudiar el dominio del problema: se debe conocer la terminología básica del dominio del problema, evitando que el cliente tenga que explicar términos que para él son obvios.
- Seleccionar a las personas a las que se va a entrevistar: se debe minimizar el número de entrevistas a realizar, por lo que es fundamental seleccionar a las personas a entrevistar. El orden de realización de las entrevistas también es importante. Normalmente se aplica un enfoque top-down, comenzando por los directivos, que pueden ofrecer una visión global, ayudar a determinar los objetivos y reducir ciertas reticencias en sus subordinados, y terminando por los futuros usuarios, que pueden aportar información más detallada.
- Determinar el objetivo y contenido de las entrevistas: fijar el objetivo que se pretende alcanzar y determinar previamente su contenido.
- Planificar las entrevistas: la fecha, hora, lugar y duración de las entrevistas deben fijarse teniendo en cuenta siempre la agenda del entrevistado.

Realización de entrevistas

Dentro de la realización de las entrevistas se distinguen tres etapas, tal como se expone en [Piattini]:

- Apertura: el entrevistador debe presentarse e informar al entrevistado sobre la razón de

la entrevista, qué se espera conseguir, cómo se utilizará la información, la mecánica de las preguntas, etc.

- Desarrollo: la entrevista en sí no debería durar más de dos horas, distribuyendo el tiempo en un 20% para el entrevistador y un 80% para el entrevistado.
- Terminación: se debe recapitular sobre la entrevista para confirmar que no ha habido confusiones en la información recogida.

Análisis de las entrevistas

Una vez realizada la entrevista es necesario leer las notas tomadas, pasarlas a limpio, reorganizar la información, contrastarla con otras entrevistas o fuentes de información, etc.

3.1.2 Brainstorming o lluvia de ideas

El brainstorming o tormenta de ideas es una técnica de reuniones en grupo cuyo objetivo es la generación de ideas en un ambiente libre de críticas o juicios [Raghavan].

Este método comenzó en el ámbito de las empresas, aplicándose a temas tan variados como la productividad, la necesidad de encontrar nuevas ideas y soluciones para los productos del mercado, encontrar nuevos métodos que desarrollen el pensamiento creativo a todos los niveles, etc. Pero pronto se extendió a otros ámbitos, incluyendo el mundo de desarrollo de sistemas; básicamente se busca que los involucrados en un proyecto desarrollen su creatividad, promoviendo la introducción de los principios creativos.

Las sesiones de brainstorming suelen estar formadas por un número de cuatro a diez participantes, uno de los cuales es el jefe de la sesión, encargado más de comenzar la sesión que de controlarla.

Como técnica de elicitación de requerimientos, el brainstorming puede ayudar a generar una gran variedad de vistas del problema y a formularlo de diferentes formas, sobre todo al comienzo del proceso de elicitación, cuando los requerimientos son todavía muy difusos.

3.1.3 Prototipos

Durante la actividad de extracción de requerimientos, puede ocurrir que algunos requerimientos no estén demasiado claros o que no se esté muy seguro de haber entendido correctamente los requerimientos obtenidos hasta el momento, todo lo cual puede llevar a un desarrollo no eficaz del sistema final.

Entonces, para validar los requerimientos hallados, se construyen prototipos. Los prototipos son simulaciones del posible producto, que luego son utilizados por el usuario final, permitiéndonos conseguir una importante retroalimentación en cuanto a si el sistema diseñado con base a los requerimientos recolectados le permite al usuario realizar su trabajo de manera eficiente y efectiva.

El desarrollo del prototipo comienza con la captura de requerimientos. Desarrolladores y clientes se reúnen y definen los objetivos globales del software, identifican todos los requerimientos que son conocidos, y señalan áreas en las que será necesaria la profundización en las definiciones. Luego de esto, tiene lugar un “diseño rápido”. El diseño rápido se centra en una representación de aquellos aspectos del software que serán visibles al usuario (por ejemplo, entradas y formatos de las salidas). El diseño rápido lleva a la construcción de un prototipo.

3.1.4 Casos de Uso

Los casos de uso son una técnica para la especificación de requerimientos funcionales propuesta inicialmente en [Jacobson] y que actualmente forma parte de la propuesta de UML [Booch99].

Un caso de uso es la descripción de una secuencia de interacciones entre el sistema y uno o más actores en la que se considera al sistema como una caja negra.

Los casos de uso son una técnica para especificar el comportamiento de un sistema: “Un caso de uso es una secuencia de interacciones entre un sistema y actores que usan alguno de sus servicios” [Schneider].

Los actores son personas u otros sistemas que interactúan con el sistema cuyos requerimientos se



están describiendo. Un actor puede participar en varios casos de uso y un caso de uso puede estar relacionado con varios actores.

Los casos de uso presentan ciertas ventajas sobre la descripción meramente textual de los requerimientos funcionales [Firesmith97], ya que facilitan la elicitación de requerimientos y son fácilmente comprensibles por los clientes y usuarios. Además, pueden servir de base a las pruebas del sistema y a la documentación para los usuarios.

3.1.5 Win Win

Es una técnica de elicitación de requerimientos basada en la recolección a través de una herramienta de soporte. Esta técnica da resultados positivos tanto al desarrollador como al usuario y ayuda en la determinación de prioridades.

Establece objetivos y procedimientos para los equipos integrados de productos - qué es lo que deben tratar de hacer los participantes? Cómo deberían proceder? Cómo sabrán cuando hayan terminado?

La condición fundamental de éxito de esta técnica es, que el proyecto será exitoso si y solo si se crean “ganadores” con cada uno de los stakeholders [Sommerville:1997]. Las principales características documentadas de la técnica Win-Win son:

- Acentúa la necesidad de conseguir un aporte de cada una de las partes implicadas.
- La aceptación del sistema es una resolución muy probable.
- Se facilita la resolución de conflictos a nivel social.
- La evolución de los requerimientos es activa.

Los pasos a seguir para llevar adelante esta técnica son los siguientes.

- Identificar los stakeholders que sean críticos para el éxito del proyecto.
- Identificar las condiciones de ganancia de los stakeholders, las cuales representan los

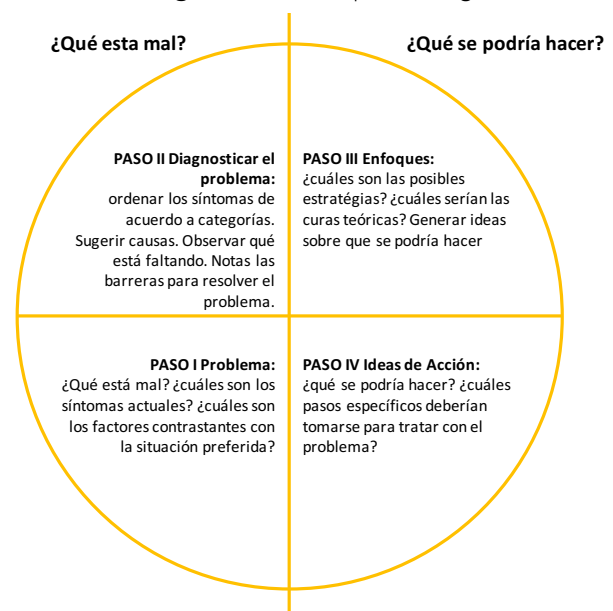
requerimientos específicos de sistema.

- Los ingenieros de requerimientos identifican problemas de conflicto en las condiciones de ganancia, y guían los stakeholders involucrados durante la negociación de compromiso.

Las condiciones de ganancia no conflictivas y el compromiso de los stakeholders son propuestas como una especificación de sistema.

- Negociar los acuerdos de Win-Win de alto nivel.
- Agrupar los acuerdos de Win-Win.
- Desarrollar los puntos anteriores hasta que el producto esté desarrollado completamente. Como se observa en la Fig. 1

Fig.1 Win Win pasos a seguir



3.2 Herramientas Automatizadas para IR

En el desarrollo de software se cuenta con una ventaja proporcionada por las herramientas CASE. Las herramientas CASE (Ingeniería del Software Asistida por Computadora) se le conoce a todo aquel software que es usado para ayudar a las actividades del proceso de desarrollo del software, en donde se ubica la ingeniería de requerimientos,

que se ha venido tratando en este artículo. Estas herramientas se concentran en capturar requerimientos, administrarlos y producir una especificación de requisitos.

Existen muchas y muy variadas herramientas CASE que pueden ser utilizadas por los desarrolladores de software en sus proyectos, y de la forma más conveniente para ellos. Si es importante hacer ver que estas herramientas fungan como un medio facilitador para agilizar y mejorar los procesos involucrados en todo el ciclo de vida presentado por la IR, y que en conjunto ayudan a la construcción final de un producto de software terminado.

Estas herramientas permiten entre otras cosas tener un mayor control en proyectos complejos, reducir costos y retrasos en los proyectos, ayudan a determinar la complejidad y los esfuerzos necesarios.

A continuación se presentan actividades y técnicas, que no pertenecen a un modelo de proceso en sí, sino, que son una alternativa al material publicado por diferentes autores y que, desde mi punto de vista, son las más importantes y posibles de ser consideradas a la hora de enfrentar un proceso de desarrollo sea este pequeño, mediano o grande.

3.2.1 RQA

TCP Sistemas e Ingeniería, desarrolló en el 2006 la nueva versión de la herramienta IRQA 3.5.1; soporta los procesos de recolección, análisis y construcción de especificación de requisitos (TCP, 2008).

3.2.2 RequisitePro

IBM Rational Software en el 2006 saca al mercado la nueva versión Requisite Pro, herramienta que permite que los requisitos se encuentren documentados bajo estándares recomendados por IEEE, ISO, CMM y RUP, entre otros (IBM, 2008).

Esta herramienta se integra con aplicaciones para la administración de cambios, herramientas de modelado de sistemas y con herramientas de pruebas. Esta integración asegura que los diseñadores conocen los requerimientos del

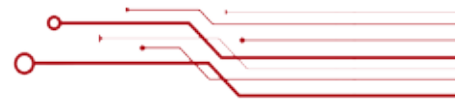
usuario, del sistema y del software en el momento de su desarrollo.

El desarrollo de software es una tarea de equipo, de tal forma, es crítico que todos los miembros del equipo posean un entendimiento compartido de la visión de sus proyectos, metas, especificaciones y requerimientos; pero, ¿cómo puede conseguirse cuando los equipos se encuentran geográficamente distribuidos y funcionalmente aislados, no pudiendo comunicarse entre sí en tiempo y forma? La solución a esta necesidad es IBM Rational RequisitePro. IBM Rational RequisitePro es una solución fácil de usar, es una herramienta de administración de requerimientos que le permite al equipo crear y compartir sus requerimientos utilizando métodos familiares basados en documentos potenciados por la aplicación de las capacidades de una base de datos, tales como la trazabilidad y análisis de impacto.

3.2.3 DOORS

Telelogic desarrolló la herramienta para administración de requisitos DOORS; esta herramienta permite capturar, analizar y administrar un rango de información para asegurar el cumplimiento del proyecto en cuanto a requisitos creada por Quality Systems and Software. Esta herramienta permite capturar, relacionar, analizar y administrar un rango de información para asegurar el cumplimiento del proyecto en materia de requerimientos. DOORS permite el acceso de un gran número de usuarios concurrentes en la red, manteniendo en línea un gran número de requerimientos así como su información asociada. DOORS ayuda al usuario a procesar las solicitudes de cambios de requerimientos en línea. Permite realizar cualquier modificación vía remota cuando la base de datos está off-line, incorporando sus actualizaciones a la base de datos maestra. Esto hace más fácil la comunicación del equipo con otras organizaciones, subcontratistas y proveedores.

Esta herramienta proporciona rastreabilidad multi-nivel para aquellas relaciones entre requerimientos que poseen gran tamaño. DOORS cuenta con un wizard que le permite generar enlaces a reportes de muchos niveles, para desplegarlos en la misma



vista.

Beneficios de DOORS

- Análisis y comparación de requerimientos.
- Clasificación de requerimientos.
- Interpretación manual de cada requerimiento.
- Identificación de Inconsistencias.
- Operación vía batch.
- Permite compartir requerimientos entre proyectos.
- Permite crear relaciones entre requerimientos mediante la táctica drag-and-drop
- Envía una notificación vía email cuando los cambios son revisados.
- Permite visualizar los cambios pendientes de otros usuarios para anticipar el impacto que ocasionará.
- Despliega estadísticas y métricas a través de gráficas.
- Los documentos están escritos en lenguaje claro, lo que proporciona una comprensión inmediata de cada requerimiento.
- Permite importar sus documentos a formatos de herramientas de Microsoft Office, RTF, HTML, texto, entre otros.
- Las plantillas presentan la información de manera estandarizada.

CONCLUSIONES

La evolución de los estudios encarados por la Ingeniería de Requerimientos se fue dando paulatinamente. Sin embargo, a partir de los 90, los esfuerzos se concentraron en la búsqueda de técnicas, métodos y herramientas que pudieran ser aplicados durante el proceso de definición de requerimientos para arribar a una etapa de diseño exitosa, dejando de lado la obtención de una metodología capaz de adaptarse a cualquier tipo de sistema y paradigma, brindando un marco

de trabajo referencial, independiente del método a aplicar.

Es muy importante mencionar que el poder formular una especificación de requerimientos completa y consistente, es un paso muy importante para evitar cometer errores en la definición de los requerimientos, ya que los mismos pueden resultar muy caros de corregir una vez desarrollado el sistema. De ahí, la vital importancia que tiene la ingeniería de requerimientos en generar una adecuada especificación que contemple claramente y sin ambigüedades los requerimientos del sistema a desarrollar, con el fin primordial de evitar que los proyectos fracasen debido a una mala elaboración de la definición y especificación de requerimientos.

Sabiendo la importancia que tiene la Ingeniería de Requerimientos, no se le preste la debida atención a esta actividad tan importante para el proceso de desarrollo de software. A pesar del aporte de varios autores respecto al tema, todavía queda muchos aspectos que trabajar de forma tal que logremos integrar adecuadamente los requerimientos funcionales y no funcionales, la evaluación de especificaciones alternativas y otras.

Sabemos que cada actividad y técnica de la IR utilizada individualmente, dará diferentes soluciones para diferentes proyectos, incluyendo aquellos casos en los que el dominio y el área del problema son el mismo. Por esta razón, se considera que no existe un modelo de proceso ideal para la IR; encontrar el método o la técnica perfecta es una ilusión, pues cada método y técnica ofrece diferentes soluciones ante un problema, sin embargo no podemos dejar de trabajar aportando en este aspecto, de forma que se logre en algún momento uniformizar el proceso de la IR.

La ingeniería de requerimientos es una actividad compleja que requiere un proceso definido que debe ser bien planificado y monitoreado. Se centra principalmente en la extracción de información, desde múltiples y diversas fuentes, y además involucra la organización de la información relacionada al problema que necesita ser resuelto. Consiste de una secuencia de etapas: elicitación, análisis, especificación, validación y administración de requerimientos y la aplicación de una serie de

técnicas y herramientas que permitan obtener un documento robusto que guíe el proceso de desarrollo de la forma más eficiente y eficaz.

REFERENCIAS BIBLIOGRÁFICAS

Avionics Software Engineering. Requirements, [1998] Management Guidebook. Software Engineering Management Working Group. 1998

Borland. Mitigating Risk With Effective Requirements Engineering NG How to improve decision-making and opportunity through effective requirements engineering. Part two in a series about understanding and managing risk. April 2005

Booch, Grady, Jacobson, Ivar y Rumbaugh, James. El Proceso Unificado de Desarrollo de Software, España: Pearson Educación, 2007. 688 p

Brackets. John W. Boston University. Software Requirements. SEI Curriculum Module SEI-CM-19-1.2. Enero 1990.

Hans van Vliet, "Software Engineering: Principles and Practice", segunda edición, John Wiley & Sons, 2001.

Kotonya G. 1995 and I. Sommerville, "Requirements Engineering with Viewpoints", Software Eng. Journal 1996; 11(1): 5-11.

Niessink Frank "Software Requirements"

Sommerville Ian, 2005, "Ingeniería del Software", Séptima edición, México DF, Editorial Pearson.

Pressman, Roger S. 2006, "Ingeniería del Software: Un enfoque práctico", Sexta edición, México DF, Editorial McGraw Hill.

Swebok Software Engineering Body of Knowledge, 2004.

Schneider, G., Winters, J.P., Applying Use Cases, Second Edition, Addison-Wesley, Massachusetts, USA, 2001

Sommerville I. Ingeniería del Software, setp. Edición. Madrid 2005

TCP Sistemas e Ingeniería. IRQA Integral Requirement Analyser.

TELELOGIC AB. Gestión de requisitos para equipos en colaboración. Disponible en <http://www.telelogic.es/products/doors/index.cfm>.

Páginas Web

IEEE Std 610.12-1990, "IEEE Standard Glossary of Software Engineering Terminology", http://standards.ieee.org/reading/ieee/std_public/description/se/610.12-1990_desc.html

<ftp://ftp.comp.lancs.ac.uk/pub/reports/1995/CSEG.10.95.ps.Z>

<http://www.irqaonline.com/>.

<http://www.monografias.com/trabajos-pdf5/diagrama-caso-uso/diagrama-caso-uso.shtml#ixzz4M1t5DqHE>

