

Gestión y tecnología para la ingeniería de requerimientos en servicios computacionales

Management and technology for engineering requirements in computer services

Engenharia de gestão e requisitos de tecnologia em serviços de informática

Luis Ambrosio Velázquez García

Universidad Autónoma de Querétaro, México

luismxbon@hotmail.com

Resumen

El presente artículo describe el enfoque teórico sobre la gestión en la ingeniería de requerimientos que toda organización necesita a través de la incorporación de la tecnología, para la toma de decisiones sobre qué método o modelo se empleará, ya que desempeña un papel primordial en el proceso de producción de software dado que se enfoca en un área fundamental: la definición de lo que se desea producir.

La 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. Debido a ello en este trabajo se exponen dichos métodos o modelos con sus características, técnicas y/o herramientas a tomar en cuenta para el buen desarrollo de software y su implementación.

Esta investigación es documental y se concluyó que un factor relevante en la gestión y tecnología para la ingeniería de requerimientos es la viabilidad de implementar un software mediante el análisis de requerimientos y su especificación formal, lo cual reducirá costos y problemas en la operación.

Palabras clave: gestión industrial, elección de tecnología, aplicación informática.

Abstract

This article describes the theoretical approach on the management in Requirements Engineering that every organization needs through the incorporation of technology, for decision making about which method or model will be used, Since it plays an essential role in the process of software production given that focuses on a fundamental area: the definition of what you want to produce.

The main task consists in the generation of correct specifications describing clearly, unambiguously, in way consistent and compact, the needs of users or customers; in this way is intended to minimize the problems of poor requirements management in systems development. As a result in this work are exposed such methods or models with their characteristics, techniques and tools to consider for good software development and implementation.

This documentary research concluded that a relevant factor in management and technology for requirements engineering It is the feasibility of implementing a software requirements analysis and its formal specification, which will reduce costs and problems in operation.

Key Words: industrial management, choice of technology, computer application.

Resumo

Este artigo descreve a abordagem teórica da engenharia de requisitos de gestão que toda organização precisa através da introdução de tecnologia, para tomar decisões sobre qual método ou modelo será usado como ela desempenha um papel fundamental no processo de produção de software, uma vez que se concentra em uma área chave: a definição do que deve ser produzido.

A tarefa principal consiste em gerar especificações corretas que descrevem claramente, de forma inequívoca, coerente e compacto, as necessidades dos usuários ou clientes; Esta abordagem destina-se a minimizar os problemas de má gestão dos requisitos no desenvolvimento do sistema. Como resultado deste trabalho, esses métodos ou modelos com as suas características, técnicas e / ou ferramentas para ter em conta para o bom desenvolvimento de software e sua implementação são discutidas.

Esta pesquisa é documental e concluiu que um fator relevante na engenharia de requisitos de gestão e tecnologia é a viabilidade de implementação de um software de análise de requisitos e especificação formal, o que reduzirá os custos e problemas operacionais.

Palavras-chave: gestão industrial, escolha da tecnologia, aplicação de computador.

Fecha recepción: Diciembre 2015

Fecha aceptación: Julio 2016

Introducción

Los requisitos son la base de todos los proyectos que involucren el desarrollo de software. La definición de los usuarios, clientes, proveedores, desarrolladores y empresas involucradas en la creación de un nuevo sistema, dará como resultado lo que el sistema debe de hacer para satisfacer las necesidades de dicho proyecto.

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 en los requerimientos.

Al acordar los requisitos para el sistema, se proporcionan las bases para la planificación del desarrollo de un proyecto de software. Estos deben ser entendidos por todos los involucrados en el proyecto, expresados en un lenguaje común, captando la necesidad del problema por completo, sin ambigüedades.

Los requisitos permiten la gestión de los riesgos, inclusive desde que se inicia el desarrollo del sistema. Los riesgos planteados pueden ser rastreados, su impacto evaluado y los efectos del plan mitigados mediante grandes ahorros en el proyecto.

La ingeniería de requerimientos sirve para determinar las necesidades que un cliente tiene en la implementación o desarrollo de software, mediante el proceso de recopilar, analizar y verificar la especificación de los requisitos del software de una manera completa y correcta (Chaves, 2011). Además, contribuye a construir un producto de software de alta calidad, bajo las restricciones de tiempo y presupuesto, donde se requiere de rigor, creatividad, documentación y gestión en todas sus actividades (Pohl, 2013).

Los requerimientos son fundamentales en los proyectos que involucren el desarrollo de software, pues abarcan actividades como la planeación, que se refiere a las estimaciones de tiempos y costos; asimismo consideran la definición de los recursos necesarios y la elaboración de cronogramas, que será uno de los principales mecanismos de control con los que se cuenta durante la etapa de desarrollo.

Generalmente los costos que representa el desarrollo de software son a menudo mayores que el hardware. La Ingeniería de Requerimientos (IR) tiene que ver con que el desarrollo se realice de tal manera que sea económicamente viable. Cuando los clientes, usuarios y desarrolladores de software no conocen de forma precisa el problema que quieren resolver, lo más probable es que no se obtenga una solución útil y óptima.

Es necesario conocer y obtener los requisitos de manera correcta, ya que son un punto crítico en el desarrollo del software, independientemente del tipo de desarrollo que se trate. Una mala percepción o documentación de los mismos llevaría a una gran cantidad de problemas que surgen en el ciclo de vida del desarrollo.

Existen técnicas, herramientas y modelos que tienen un papel primordial en el proceso de producción de software ya que se enfocan en 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 esa manera se pretende minimizar los problemas relacionados por la mala gestión de los requerimientos en el desarrollo de sistemas.

Debido a lo anterior este trabajo pretende exponer las técnicas, herramientas y modelos a tomar en cuenta y proponer un buen desarrollo de software.

MARCO TEÓRICO

Conceptos básicos de la Ingeniería de Requerimientos

La perspectiva teórica de esta investigación retoma los principios de los estudios realizados por Leite (1987), Sommerville (2005), Arias (2005), Jeffries (2001), Insfran (2014), entre otros, donde describen algunas de las definiciones más reconocidas sobre la Ingeniería de Requerimientos (IR).

Por ejemplo, para Leite (1987) la ingeniería de requerimientos es el proceso mediante el cual se intercambian diferentes puntos de vista para recopilar y modelar lo que el sistema va a realizar. Este proceso utiliza una combinación de métodos, herramientas y actores, cuyo producto es un modelo del cual se genera un documento de requerimientos.

O bien una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo, o una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal (IEEE 1990).

Para Sommerville et al. (2005) es un requerimiento o simplemente una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de este.

Arias (2005) establece que los requerimientos son fundamentales en los proyectos que involucren el desarrollo de software. Abarcan actividades como la planeación, que se refiere a las estimaciones de tiempos y costos, también consideran la definición de los recursos necesarios y la elaboración de cronogramas que será uno de los principales mecanismos de control con los que se cuenta durante la etapa de desarrollo.

Jeffries et al. (2001), comentan que una historia no es solo una descripción de una funcionalidad, normalmente expuesta en un post-it, sino que una historia de usuario además está formada por otras dos partes:

1. La conversación que conlleva, ya que es una herramienta para interactuar; y
2. La manera como se confirman su implementación, pruebas y verificación.

Los requerimientos funcionales son los que definen las funciones que el sistema será capaz de realizar, ya que describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Es importante que se describa el qué y no el cómo se deben hacer esas transformaciones. Estos requerimientos, al mismo tiempo que avanza el proyecto de software, se convierten en los algoritmos, la lógica y gran parte del código del sistema (Insfran, 2014).

Las historias de usuario se utilizan para definir los requerimientos de un sistema de software y también para crear estimaciones para la planeación de las interacciones. Estas son escritas por los clientes en forma de las cosas que quieren que el sistema haga por ellos (Jeffries et al., 2001).

Dichas tarjetas son una técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. A su vez los requerimientos de software pueden dividirse en 2 categorías: 1) requerimientos funcionales y 2) requerimientos no funcionales (Ibíd., 2001).

Los requerimientos permiten verificar si se alcanzaron o no los objetivos que se establecen en el proyecto y además son un estándar de las necesidades de los clientes o usuarios del sistema, ya que se va a estar verificando si se están cumpliendo las metas trazadas (Arias, 2005).

Insfran (2014) menciona que es importante saber que un requerimiento debe ser:

- Especificado por escrito, es decir como todo contrato o acuerdo entre dos partes.
- Posible de probar o verificar, ya que si un requerimiento no se puede comprobar, entonces ¿cómo se sabe si se cumplió con él o no?
- Conciso, un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- Completo, si el requerimiento está completo, si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.
- Consistente, es decir, un requerimiento es consistente si no se contradice con otro requerimiento.
- No ambiguo, por ejemplo, un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición no debe causar confusiones al lector.

Durante la etapa de especificación de requerimientos se pueden presentar muchos inconvenientes, los cuales son importantes de identificar y prevenir. A continuación se presenta un listado que Sommerville et al. (2005) ofrece sobre los problemas más comunes en este proceso:

- Los requerimientos no son obvios y vienen de muchas fuentes.
- Son difíciles de expresar en palabras (el lenguaje es ambiguo).
- La cantidad de requerimientos en un proyecto puede ser difícil de manejar.
- Un requerimiento puede cambiar a lo largo del ciclo de desarrollo.
- El usuario no puede explicar lo que hace.
- Tiende a recordar lo excepcional y olvidar lo rutinario.
- Habla de lo que no funciona.
- Los usuarios tienen un vocabulario distinto al de los desarrolladores.

Modelos, métodos y/o herramientas que describen la Ingeniería de Requerimientos

A continuación se exponen los diferentes modelos, métodos y/o herramientas que describen la Ingeniería de Requerimientos en las organizaciones. Existen varias alternativas para poder definir las características óptimas con las que debe de contar una buena tarjeta de historia de usuario. Algunas de estas son:

Las 3 “C”

Una historia de usuario describe la funcionalidad deseada desde la perspectiva del cliente, que describe la funcionalidad, quién la necesita, cómo y porqué se va a utilizar. Las características básicas de una historia de usuario se pueden resumir en tres elementos que se conocen como las 3 “C” de una historia de usuario por sus iniciales en inglés *Card*, *Conversation*, *Confirmation* (Jeffries, 2001):

1. Tarjeta (*Card*): es la descripción escrita de la historia, que sirve como identificación, recordatorio y también ayuda a planificar.
2. Conversación (*Conversation*): es el núcleo de la historia, el diálogo que ocurre con los usuarios, notas, grabaciones, prototipos y documentos.
3. Confirmación (*Confirmation*): es el criterio de las pruebas de aceptación que el usuario va a utilizar para confirmar que la historia fue terminada.

Otra característica importante la representan los acrónimos INVEST y SMART (por sus siglas en inglés) y que desde hace tiempo son marcados como una referencia para la realización de una buena tarjeta de historia.

El modelo INVEST

Wake (2003) sugirió el modelo INVEST, es decir, una historia Independiente, Negociable, Estimable, Pequeña y Comprobable. Dichos elementos se describen a continuación:

- Independiente (*Independent*): una historia debería ser independiente de otras. Las dependencias entre las historias hace que sea más difícil planificar, priorizar y estimar. Aunque no siempre se puede realizar esto, se pueden reducir las dependencias haciendo una combinación de historias, o partiendo historias de forma diferente.

- Negociable (*Negotiable*): una historia de usuario es negociable. No es un contrato explícito de las características que debe contener el desarrollo del sistema. La tarjeta de la historia es tan solo una descripción corta que no incluye detalles.
- Valiosa (*Valuable*): cada historia tiene que tener valor para el cliente. Una forma muy buena de generar historias valiosas es hacer que las escriba el cliente.
- Estimable (*Estimable*): los desarrolladores necesitan poder estimar una historia de usuario para permitir que se pueda priorizar y planificar la historia. No se necesita una estimación exacta, sino solo lo suficiente para ayudar a las peticiones de los clientes y planificar la implementación de lo que está en la tarjeta de historia. También se involucra una función del tamaño; historias más grandes son más difíciles de estimar.
- Pequeña (*Small*): una buena historia debe ser pequeña en esfuerzo, generalmente representando no más de 2-3 personas/semana de trabajo. Historias más pequeñas, tienden a obtener estimaciones más precisas.
- Comprobable (*Testable*): una historia necesita poder probarse para que ocurra la etapa de confirmación. Si un cliente no sabe como probar algo, esto puede indicar que la historia no es lo suficientemente clara, o que no corresponde con algo valioso para ellos, o que el cliente solo necesita ayuda en las pruebas.

Los atributos que conlleva el método INVEST, son: el ciclo de retroalimentación de la propuesta, la estimación y la implementación, que ayudan de manera significativa a la realización de las tarjetas de historia de usuario y enseñan al equipo de trabajo lo que cada uno debe de hacer para que el proyecto del desarrollo del software no fracase.

El modelo SMART

Como menciona Wake (2003), el modelo basado en Tareas Inteligentes o SMART, contempla lo específico, medible, realizable, pertinente y el tiempo de caja que debe tener una tarjeta de historia de usuario. Se define por lo siguiente:

- Especifico (*Specific*): una tarea tiene que ser lo suficientemente específica como para que todo el grupo pueda entender lo que está plasmado en ella.

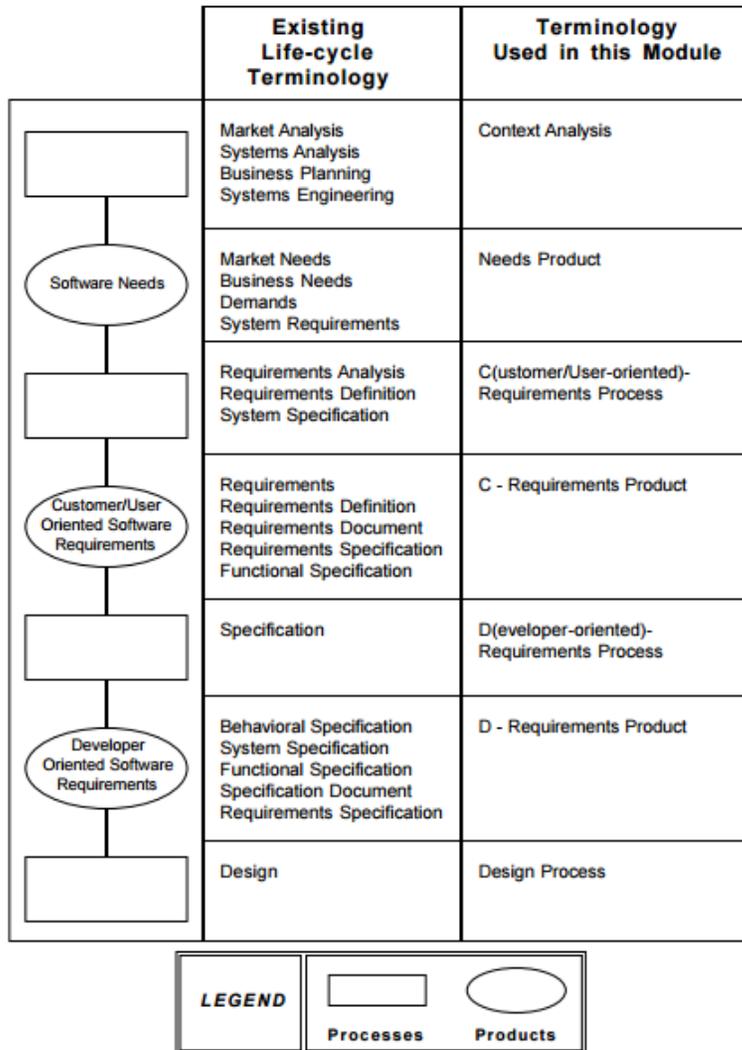
- **Mensurable (*Measurable*):** ¿se puede decir que ya está terminado? Se debe ser mensurable y entender que el equipo tiene que ponerse de acuerdo en que el proceso hace lo que se pretende, las pruebas están concluidas y el código ha sido programado.
- **Realizable (*Archiveable*):** el programador debe de ser capaz de realizar la tarea que se le pide. Existe una regla en el equipo de desarrolladores donde cualquiera puede pedir ayuda cuando lo necesite, lo cual sin duda garantiza que se concluya la tarea.
- **Relevante (*Relevant*):** cada tarea debe ser relevante y contribuir de manera importante al desarrollo del proyecto. Las historias se dividen en tareas para beneficio de los desarrolladores y al finalizarlas se explican y justifican al cliente.
- **Tiempo de caja (*Time – Boxed*):** una tarea está limitada a una duración específica. Puede ser una estimación en horas o días, pero si no se puede realizar debe existir una expectativa de cuándo pedir ayuda. Si una tarea es más complicada de lo esperado, el equipo debe saber dividir la tarea, cambiar al desarrollador o hacer algo que ayude a realizar la tarea en el tiempo estimado.

Para Brackett (1990), los elementos que se integran en la terminología del ciclo de vida que muestra la Figura 1, se dan cuando se han determinado los requisitos generales del sistema y se ha tomado la decisión de que ciertas funciones sean realizadas por el software:

- 1) El proceso de requisitos debe comenzar con definir y analizar el Contexto del que se trate, es decir realizar una breve descripción de la necesidad,
- 2) Definir la Necesidad: esta contempla los elementos externos de la empresa como son el mercado, las necesidades internas de la empresa, la demanda real y potencial, así como los requerimientos propios de sistemas que ayuden a las decisiones en la empresa,
- 3) Delimitar el proceso de requerimientos, es decir, evitar demasiados detalles, debe estar presente la flexibilidad para que el equipo pueda ajustar cuánto de la necesidad del usuario se va a implementar,
- 4) Identificar los requerimientos del producto, es ser capaces de seguir desarrollando en cualquier secuencia y que el equipo pueda utilizarlos para planificar,
- 5) Definir el proceso de requerimiento orientado hacia el desarrollador,

- 6) Delimitar los requerimientos del producto (especificación del comportamiento del usuario final y del sistema) y finalmente
- 7) El diseño del proceso que implicará analizar y comprobar los criterios de aceptación del documento o la lista para la historia al definir el ciclo de vida.

Figura 1. Terminología usada para definir el ciclo de vida



Fuente: Brackett (1990).

Conceptos básicos de gestión y tecnología

Por otro lado, es necesario definir el papel que juegan la gestión y la tecnología, para poder aplicar la ingeniería de requerimientos, por tal motivo se analizan los conceptos a partir de las aportaciones de Valencia (1996), Mora (1999), BID-SECAB-CINDA (1990), y Zoltan (1995).

En términos generales, los conceptos de administración, gerencia y gestión son sinónimos, a pesar de los grandes esfuerzos y discusiones que se han hecho para diferenciarlos. En la práctica se observa que el término management ha sido traducido como administración, pero también como gerencia.

En esa concepción, le corresponde al gerente echar una mirada al entorno de modo que la organización pueda generar desarrollo: tomar recursos y producir más recursos. Al administrador le corresponde más el mantenimiento y conservación, mientras que a la administración se le concibe funcional o vertical (Valencia, 1996).

Mora (1999) planteó que la gestión es lineal o tradicional donde es sinónimo de administración. Por gestión se entiende el conjunto de diligencias que se realizan para desarrollar un proceso o para lograr un producto determinado. Se asume como la dirección y el gobierno de actividades para hacer que las cosas funcionen, con capacidad para generar procesos de transformación de la realidad.

Por otro lado, el BID-SECAB-CINDA (1990) ofrece la siguiente clasificación de la tecnología de acuerdo con el grado de incorporación y el grado de modernidad:

- a) Grado de Incorporación:
 - Hardware: es la tecnología incorporada en máquinas.
 - Software: es la tecnología no incorporada y se presenta a través de revistas, libros, manuales, videos, programas de computador.
 - Orgware: estructuras organizacionales.
 - Humanware: es la incorporada en personas, quienes tienen un "know how".
- b) Grado de modernidad:
 - Tecnología primitiva: corresponde a las épocas primitivas, esclavistas y feudales.

- Tecnología moderna: en contraste con la primitiva es la producida en los últimos decenios.
- Tecnología atrasada: es aquella "que ha sido superada en algún factor", por ejemplo, la máquina de escribir eléctrica con relación con la de escribir con memoria.
- Tecnología de punta: es aquella que acaba de ser producida. Está "recién sacada del horno".

Al ser multi e interdisciplinaria, la gestión tecnológica no puede ser objeto propio de ninguna profesión o disciplina (Zoltan, 1995). Es un proceso que se ocupa de las interfaces entre la ciencia, la ingeniería, la economía y la gerencia de instituciones.

La gestión tecnológica promueve la organización y la ejecución de tareas en relación estrecha con los agentes (investigadores, ingenieros, científicos, tecnólogos), pero no es en sí, ni contiene en sí la realización misma de la investigación, la innovación o el control de calidad (Ibíd., 1995).

En el glosario de términos del BID-SECAB-CINDA (1990), se indica que la gestión tecnológica es la disciplina en la que se mezclan conocimientos de ingeniería, ciencias y administración con el fin de realizar la planeación, el desarrollo y la implantación de soluciones tecnológicas que contribuyan al logro de los objetivos estratégicos y técnicos de una organización.

De acuerdo con Zoltan (1995), la gestión tecnológica se da a nivel de un país. Dado que este se encuentra en el mercado mundial de la tecnología, los clientes extranjeros de las empresas nacionales pero también la competencia encuentran políticas de gobierno, así como un conjunto de reglas internacionales sobre comercio, inversión, calidad, ambiente, patentes, entre otros.

Por otro lado, dentro del país se encuentran: un gobierno con políticas de ciencia, tecnología y educación; un mercado nacional de tecnología y finalmente las empresas que tienen clientes y competidores y en todo el contexto nacional y mundial deben tomar una serie de decisiones sobre la tecnología. Deben decidir si hacen un desarrollo propio o si buscan la tecnología en el exterior; en el primer caso deben hacer innovación, en el segundo evaluar, seleccionar, negociar, comprar,

adaptar, entre todas esas actividades que exigen todo un concepto de planeación estratégica (Ibíd., 1995).

METODOLOGÍA

A continuación se presenta el objetivo general y objetivos específicos, así como la hipótesis y el método para ofrecer un panorama general para describir el proceso de la investigación.

Objetivo general

Plantear un procedimiento de la Ingeniería de Requerimientos a través de la Gestión y la Tecnología, que permitan comparar los requisitos, técnicas, herramientas y modelos para su aplicación en una empresa de servicios computacionales, buscando satisfacer las necesidades de las condicionantes que comúnmente se omiten en el desarrollo de los requerimientos.

Objetivos específicos

- Obtener información documental de los requisitos, técnicas, herramientas y modelos de la Ingeniería de Requerimientos.
- Realizar el comparativo de las características.
- Evidenciar cuáles son las ventajas, desventajas, importancia y alcance.

Hipótesis

Se pretende esbozar un procedimiento de la Ingeniería de Requerimientos, caracterizado por su importancia y alcance hacia el usuario, considerando los mejores resultados de los enfoques examinados y apoyándose en diversos métodos, técnicas y herramientas ya desarrollados. Además se busca unificar la terminología empleada en el campo de la Ingeniería de Requerimientos, eliminando las diferencias que puedan surgir como consecuencia de confusiones semánticas y así mejorar el proceso de la misma.

Por lo anterior, se plantea la siguiente hipótesis de investigación:

H_i: SI una empresa de servicios computacionales aplica la Ingeniería de Requerimientos, ENTONCES se logrará eficiencia en el proceso.

H₀: SI una empresa de servicios computacionales no aplica la Ingeniería de Requerimientos, ENTONCES no habrá eficiencia en el proceso.

Método

La investigación se constituye esencialmente en el enfoque cualitativo-documental, dado que la población y muestra está integrada por documentos y se constituye en una estrategia donde se observa y reflexiona sistemáticamente sobre realidades usando para ello diferentes tipos de documentos (Galán, 2011). La población documental está conformada por referencias de uso y conocimiento científico universal, básicamente de artículos científicos especializados nacionales e internacionales. Indaga, interpreta, presenta datos e informaciones sobre un tema determinado de cualquier ciencia, utilizando para ello el análisis y cuyo fin es obtener resultados que puedan servir como base para el desarrollo de la creación científica (Cortés, 2004).

Discusión y conclusiones

Es necesario priorizar, definir e implementar en los procesos de la ingeniería de requerimientos un proceso formal, con actividades y productos claramente definidos, donde también se encuentre un documento que contenga los resultados de esta etapa para el correcto desarrollo del proyecto a implementar.

Cuando no existe un documento que contenga los procesos formales, no se pueden contemplar correctamente los costos, los tiempos de desarrollo, los riesgos y las estimaciones que van surgiendo durante la vida del proyecto, y por consiguiente los requerimientos no se concretan y el producto que se quiere obtener fracasa o no cumple con las expectativas primordiales del cliente y del desarrollador mismo.

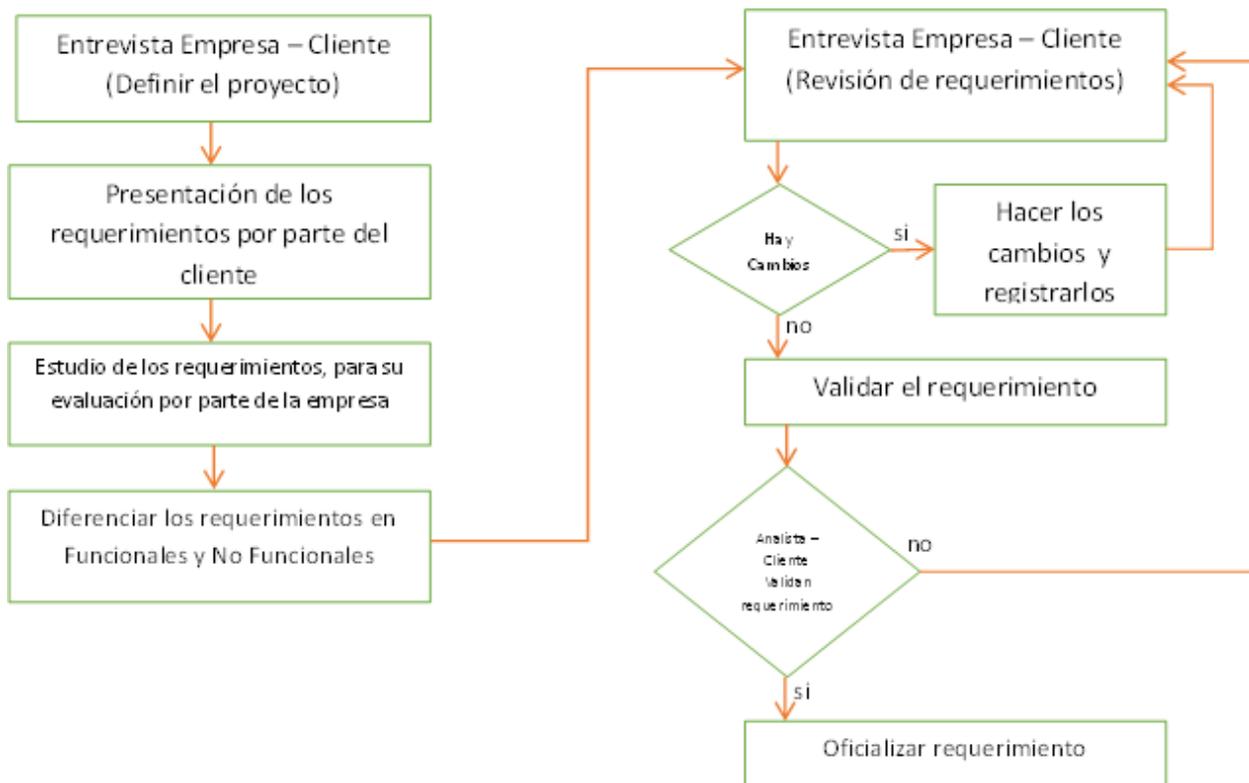
Se requiere de la documentación y mantenimiento de los requerimientos para un producto de software determinado tomando en cuenta el aporte de la Ingeniería de Requerimientos.

Así como la determinación de la viabilidad de llevar a cabo el software mediante la obtención y análisis de requerimientos y su especificación formal.

Es necesario definir el procedimiento de la Ingeniería de Requerimientos a través de la Gestión y la Tecnología como elemento básico para el cumplimiento de los objetivos marcados en cualquier proyecto para implementar en la organización. La fase de la validación del procedimiento propuesto, se debe realizar mediante la aplicación a un caso de estudio real aplicado a la organización mediante la aprobación del cliente.

En la siguiente figura 2 se presenta una propuesta que comienza con el procedimiento de la Ingeniería de Requerimientos para comparar los requisitos, técnicas, herramientas y modelos de aplicación en una empresa de servicios computacionales como parte del análisis realizado en la presente investigación.

Figura 2. Procedimiento de la IR



Fuente: Creación propia (2016), a partir de Arias (2005), Jeffries (2001), Insfran (2014), Baez (2001), Beck (2000), Cohn (2009), Mazan (2010) y Rzepka (1989).

Esta propuesta, generada como resultado del análisis teórico, contiene el diseño de las actividades comunes y el formato que se realizará para el proyecto, el cual deberá incorporar características de las metodologías de la ingeniería de requerimientos, buscando en todo momento satisfacer las necesidades de las condicionantes que comúnmente se omiten en el desarrollo de los requerimientos.

Se concluye que en toda organización que desee implementar a través de la gestión y la tecnología una ingeniería de requerimientos, debe en la primera entrevista (entre la empresa que ofrece el servicio y el cliente) definir el proyecto, sus expectativas y alcance; así como el impacto que tendrá dentro de la organización y hacia el exterior.

Una vez que se exponga el proyecto se deberán presentar los requerimientos del cliente, estos se analizan por parte de la empresa oferente y se determina cuáles son funcionales y no funcionales. Se deberá realizar una segunda entrevista donde se revisan los requerimientos, se diagnostican los cambios (si existieran) de cada uno de los mismos y se validan. Una vez validados por ambas partes, estos se oficializan y se comienza a trabajar en dicho requerimiento.

El uso y desempeño de un buen modelo aplicado en la Ingeniería de Requerimientos se utiliza para poder determinar y definir todas las actividades involucradas en el descubrimiento, documentación y mantenimiento de los requerimientos para el buen desarrollo de un proyecto y ayudar a determinar los procesos de viabilidad para saber si este es factible o no.

Al obtener las herramientas que permiten que el proyecto esté bien documentado, y donde tanto el cliente como el desarrollador utilizan el mismo lenguaje, se logra una especificación de manera formal, obteniendo un subproceso de validación que verifica que realmente lo que pide el cliente define el sistema que se está desarrollando (Sikora et al. 2012).

Bibliografía

- Arias, M. (2005). La Ingeniería de Requerimientos y su importancia en el desarrollo de proyectos de software. *Revista Intercedes*, Edición Digital, Volumen VI, Número 10.
- Baez M., Barba S. (2001). Metodología DoRCU para la Ingeniería de Requerimientos. Instituto Superior Politécnico “José Antonio Echeverría”, La Habana, Cuba.
- Beck K., Folwler, M. (2000). *Planning extreme programming*. USA: Addison Wesley Pub Co Inc.
- BID-SECAB-CINDA (1990). *Glosario de términos de gestión tecnológica*. Colección Ciencia y Tecnología N°. 28. Santiago de Chile.
- Brackett, J. (1990). *Software Requirements*, SEI Curriculum Module SEI-CM-19-1.2.
- Chaves, M. A. (2011). La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. *InterSedes*, 6(10).
- Cohn, M. (2009). *User stories applied for agile software development*. 13th printing. Indiana, USA: Pearson Education, Inc.
- Cortés C. M., Iglesias L.M. (2004). *Generalidades sobre Metodología de la Investigación*. ISBN: 968-6624-87-2. Editorial Universidad Autónoma del Carmen, Colección Material Didáctico: México.
- Galán, A.M. (2011). *Metodología de la Investigación*. Campus virtual UDES.
- INSFRAN et al. (2014). Requirements engineering in software product line engineering. *Requirements Engineering*, 19(4), 331-332.
- Jeffries et al. (2001). *Essential XP: Card, Conversation, Confirmation*. Consultado el 10/06/2013. Disponible en: <http://xprogramming.com/articles/expcardconversationconfirmation/>.
- Institute of Electrical and Electronics Engineers (1990). *IEEE Standard Glossary of Software Engineering Terminology*. IEEE Standard 610. 12-1990 (revision and redesignation of IEEE Std. 729-1983), New York.
- Leite, J. (1997). *Ingeniería de Requisitos*. Notas de Cátedra.
- Mazan, M. (2010). ¿Qué son “user stories” y por qué se tiene que “invertir” en ellas? Consultado el 13/06/2013 Disponible en: <http://agiland.pe/index.php?view=entry&id=10%3Aique-son-user-stories-historias-de-usuario-y-por-que-se-tiene-que-invertir-en->

ellas&option=com.lyften bloggi e &Itemid=29

- Mora, Julia (1999). "Transformación y gestión curricular". En: Memorias Seminario Taller Evaluación y Gestión Curricular, Universidad de Antioquía.
- Pohl, K. (2013). The three dimensions of requirements engineering. In Seminal Contributions to Information Systems Engineering (pp. 63-80). Springer Berlin Heidelberg.
- Rzepka, W. E. (1989). A requirements engineering tested: concept, status, and first results, proceedings of the Twenty-second Annual Hawaii International Conference of Systems Sciences. IEEE Computer Society. Consultado el 20/05/2013. Disponible en: <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=233>.
- Sikora et al. (2012). Industry needs and research directions in requirements engineering for embedded systems. Requirements Engineering, 17(1), 57-78.
- Sommerville et al. (2005). Ingeniería del Software. Séptima edición. Madrid, España: Pearson Educación, S.A.
- Valencia, Carlos (1996). Gerencia de Proyectos. Seminario para profesores U. de A.
- Wake, B. (2003). INVEST in Good Stories and SMART Task. XP123. Consultado el 15/07/13. Disponible en: <http://xp123.com/articles/invest-in-good-stories-and-smart-tasks/>
- Zoltan, Szabó. (1995). Seminario sobre Gestión Tecnológica. SENA.