

Gestión y tecnología para la ingeniería de requerimientos en servicios computacionales

Management and technology requirements engineering 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 to the management requirements engineering that every organization needs through the introduction of technology, for making decisions about which method or model will be used, as they have a key role in the process software production, since it focuses on a key area: the definition of what is to be produced.

The main task consists in generating correct specifications describing clearly, unambiguously, consistent and compact, the needs of users or customers; thus, it is intended to minimize problems by poor management of requirements in system development. That is why in this work such methods or models with their characteristics, techniques and / or tools that should be taken into account for a good software development and implementation are discussed.

This research is quantitative type, non-experimental approach, where design is transeccional and descriptive, and concluded that a relevant factor in management and technology requirements engineering is the viability of implementing a software by analyzing requirements and its formal specification that will reduce costs and operational problems.

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

Introdução

Os requisitos são a base de todos os projetos que envolvem o desenvolvimento de software. A definição de usuários, clientes, fornecedores, desenvolvedores e empresas envolvidas na criação de um novo sistema resultará em que o sistema deve fazer para atender às necessidades do projeto.

Um grande número de projetos de software falham por não fazer uma adequada gestão de definição, especificação e requisitos. Dentro de que a má administração pode ser encontrada fatores como falta de participação, requisitos incompletos e má gestão de mudança nos requisitos.

Ao concordar com os requisitos para o sistema, a base para o planejamento do desenvolvimento de um projeto de software são fornecidos. Estes devem ser entendidos por todos os envolvidos no projecto, expressa em uma linguagem comum, capturando a necessidade de o problema completamente, de forma inequívoca.

Os requisitos permitem o gerenciamento de riscos, incluindo a que começa a partir de desenvolvimento do sistema. Os riscos colocados pode ser rastreado, o seu impacto avaliado e minimizado os efeitos do grande plano de poupança pelo projeto.

engenharia de requisitos é usado para determinar as necessidades de um cliente tem no desenvolvimento de implementação ou software durante o processo de recolha, análise e

verificação da especificação dos requisitos de uma forma completa e correta (Chaves, 2011) de software. Ela também ajuda a construir uma qualidade de produto de software, sob as restrições de tempo e orçamento, onde se exige rigor, a criatividade, a documentação e gestão em todas as suas atividades (Pohl, 2013).

Os requisitos são fundamentais em projetos envolvendo desenvolvimento de software, uma vez que abrangem atividades como planejamento, referindo-se a estimativas de tempo e custo; também que considerar a definição dos recursos necessários e o desenvolvimento dos horários, que irá ser um dos principais mecanismos de controle, que é considerada durante a fase de desenvolvimento.

Geralmente, os custos que representa o desenvolvimento de software são muitas vezes maior do que o hardware. Engenharia de Requisitos (IR) tem a ver com que o desenvolvimento é feito de uma forma que seja economicamente viável. Quando os clientes, usuários e desenvolvedores de software não sei precisamente o problema que queremos resolver, as chances são de que uma solução útil e ótima é obtida.

Você precisa saber e obter os requisitos corretamente porque eles são um ponto crítico no desenvolvimento de software, independentemente do tipo de desenvolvimento em causa. A percepção errada ou a documentação do mesmo conduzir a uma série de problemas que surgem no ciclo de vida de desenvolvimento.

Existem técnicas, ferramentas e modelos que têm um papel primordial no processo de produção de software e com foco em uma área chave: a definição do que deve ser produzido. Sua tarefa principal é a geração de especificações corretas que descrevem claramente, de forma inequívoca, coerente e compacto, as necessidades dos usuários ou clientes; dessa forma é minimizar os problemas de má gestão dos requisitos no desenvolvimento do sistema.

Devido a isso este trabalho tem como objetivo mostrar as técnicas, ferramentas e modelos para estudar e propor um bom desenvolvimento de software.

TEÓRICO

Noções básicas de Engenharia de Requisitos

A perspectiva teórica desta pesquisa leva-se os princípios de estudos de Leite (1987), Sommerville (2005), Arias (2005), Jeffries (2001), Insfran (2014), entre outros, que descreve algumas das definições mais reconhecidos em Engenharia de Requisitos (IR).

Por exemplo, para Leite (1987) Engenharia de Requisitos é o processo pelo qual diferentes pontos de vista são trocados para coletar e modelar o que o sistema irá executar. Este processo utiliza uma combinação de métodos, meios e agentes, cujo produto é um modelo a partir da qual é gerado um documento de requisitos.

Ou uma condição ou necessidade de um usuário para resolver um problema ou atingir um objetivo ou uma condição ou uma capacidade que devem estar presentes em um sistema ou componentes do sistema para satisfazer um contrato, norma, especificação ou outro documento formal (IEEE 1990) .

Para Sommerville et ai. (2005) é um requisito ou apenas uma declaração de alto nível abstrato de um serviço a ser prestado pelo sistema ou uma restrição.

Arias (2005) afirma que os requisitos são essenciais em projetos envolvendo desenvolvimento de software. atividades de cobertura tais como planejamento, referentes a estimativas de tempo e custos, considere também a definição dos recursos necessários e desenvolver programações para ser um dos principais mecanismos de controle que é considerado durante a fase de desenvolvimento.

Jeffries et ai. (2001), dizer que uma história não é apenas uma descrição da funcionalidade normalmente exposta em um post-it, mas uma história de usuário também é formado por duas partes:

1. A conversa envolvido, uma vez que é uma ferramenta de interagir; e
2. A forma como a sua implementação, teste e verificação são confirmadas.

Os requisitos funcionais são aqueles que definem as funções que o sistema será capaz de executar, como eles descrevem as transformações que o sistema executa nas entradas para produzir saídas. É importante para descrever o que e não como essas mudanças devem ser feitas. Estes requisitos, enquanto o projeto de software progride, tornam-se algoritmos, lógica e grande parte do sistema (Insfran, 2014) de código.

histórias de usuários são usados para definir os requisitos de um sistema de software e também para criar estimativas para interações de planejamento. Estes são escritos pelos clientes sob a forma de coisas que eles querem que o sistema para fazer por eles (Jeffries et al., 2001).

Estes cartões são uma técnica utilizada para especificar os requisitos de software. É cartões de papel em que o cliente descreve brevemente as características que o sistema deve possuir, se os requisitos funcionais ou não funcionais. Por sua vez requisitos de software pode ser dividido em 2 categorias: 1) requisitos funcionais e 2) os requisitos não funcionais (ibid, 2001) ..

Requisitos mostrar ou não alcançar os objectivos definidos no projeto e eles são um padrão das necessidades dos clientes ou usuários do sistema, como será a verificação para ver se eles estão cumprindo as metas (Arias, 2005).

Insfran (2014) Ele menciona que é importante saber que a exigência deve ser:

- Especificado por escrito, que ou seja, qualquer contrato ou acordo entre duas partes.
- Possível testar ou verificar, pois, se o pedido não puder ser verificada, então como você sabe se ele é ou não cumpridos?
- Concise, uma exigência é conciso se é fácil de ler e compreender. A sua redacção deve ser simples e clara para aqueles que vão consultá-lo no futuro.
- Complete, se o pedido está completo, se precisar de mais detalhes na sua formulação, ou seja, se a informação suficiente é fornecido para sua compreensão.
- Consistente, ou seja, um requisito é consistente se não contradiz outros requisitos.
- Não ambígua, por exemplo, um requisito é inequívoca quando se tem apenas uma interpretação. A linguagem utilizada na sua definição não deve causar confusão para o leitor.

Durante a fase de especificação de requisitos pode apresentar muitas desvantagens, que são importantes para identificar e evitar. Aqui está uma lista que é apresentada Sommerville et al. (2005), prevê os problemas mais comuns neste processo:

- As exigências não são óbvias e vir de várias fontes.
- Eles são difíceis de expressar em palavras (a linguagem é ambígua).
- O número de requisitos em um projeto pode ser difícil de manusear.
- Um requisito pode mudar ao longo do ciclo de desenvolvimento.
- Você não pode explicar o que faz.
- Tende a se lembrar do excepcional e esquecer a rotina.
- Fale sobre o que não funciona.
- Os usuários têm diferentes desenvolvedores de vocabulário.

Modelos, métodos e / ou ferramentas que descrevem Engenharia de Requisitos

Os modelos diferentes, métodos e / ou ferramentas que descrevem os requisitos de engenharia em organizações estão expostas. Existem várias alternativas para definir as características ideais que devem ter uma boa placa de história do usuário. Algumas delas são:

Las 3 “C”

A história do usuário descreve a funcionalidade desejada a partir da perspectiva do cliente, descrevendo a funcionalidade, que precisa dele, como e por que usar. As características básicas de uma história de usuário pode ser resumido em três elementos conhecidos como os três "C" de uma história do usuário por sua sigla em Inglês *Card*, *Conversation*, *Confirmation* (Jeffries, 2001):

1. Card (Cartão) é a história escrita, que serve como identificação, lembrete e também ajuda a descrição plano.
2. Conversation (Conversação) é o centro da história, o diálogo acontece com os usuários, notas, gravações, protótipos e documentos.
3. A confirmação (Confirmação) é o critério para testes de aceitação que o usuário irá utilizar para confirmar que a história era acabado.

Outra característica importante é representado pelo acrônimo SMART INVEST e (por sua sigla em Inglês) e longo são marcados como uma referência para a realização de uma boa placa de história.

El modelo INVEST

Wake (2003) INVEST modelo sugerido, isto é, uma história independente, Negociável, estimável, Pequenas e verificáveis. Esses elementos estão descritos a seguir:

- Independente (Independent): a história deve ser independente dos outros. Dependências entre histórias faz com que seja mais difícil para planejar, priorizar e estimativa. Embora você não pode sempre fazer isso, você pode reduzir dependências por uma combinação de histórias ou histórias baseadas de forma diferente.
- Negociável (Negociável): uma história de usuário é negociável. Não é um contrato explícito deve conter características que o desenvolvimento do sistema. história de cartão é apenas uma breve descrição que não inclui detalhes.
- Valioso (Valioso): cada história tem que ter valor para o cliente. Uma boa maneira de gerar histórias valiosas é fazer o tipo de cliente.
- Estimáveis (estimável): os desenvolvedores precisam para estimar uma história de usuário para permitir que ele para dar prioridade e história plano. uma estimativa precisa é necessária, mas apenas o suficiente para ajudar a solicitações dos clientes e planejar a implementação do que é sobre a história do cartão. uma função do tamanho também está envolvido; maiores histórias são mais difíceis de estimar.
- Pequeno (Small): uma boa história deve ser pequeno em esforço, geralmente representando não mais de 2-3 pessoas / semana de trabalho. histórias menores tendem a obter estimativas mais precisas.
- Verificáveis (Testable): a história precisa ser testável confirmação estágio ocorre. Se um cliente não sabe como tentar algo, isso pode indicar que a história não é clara o suficiente, ou que não corresponda com algo valioso para eles, ou que o cliente precisa apenas ajuda nos testes.

Os atributos que carrega o método INVEST são: o ciclo de feedback da proposta, a estimativa ea implementação, que ajudam significativamente para a realização cartões maneira história do usuário e ensinar à equipe o que cada um deve fazer para o projeto de desenvolvimento de software não falha.

El modelo SMART

Como Wake (2003) menciona, o Smart baseada em tarefas ou modelo ESPERTO inclui específicos, mensuráveis, alcançáveis, relevantes e caixa de tempo deve ter um cartão de história do usuário. É definido pela seguinte:

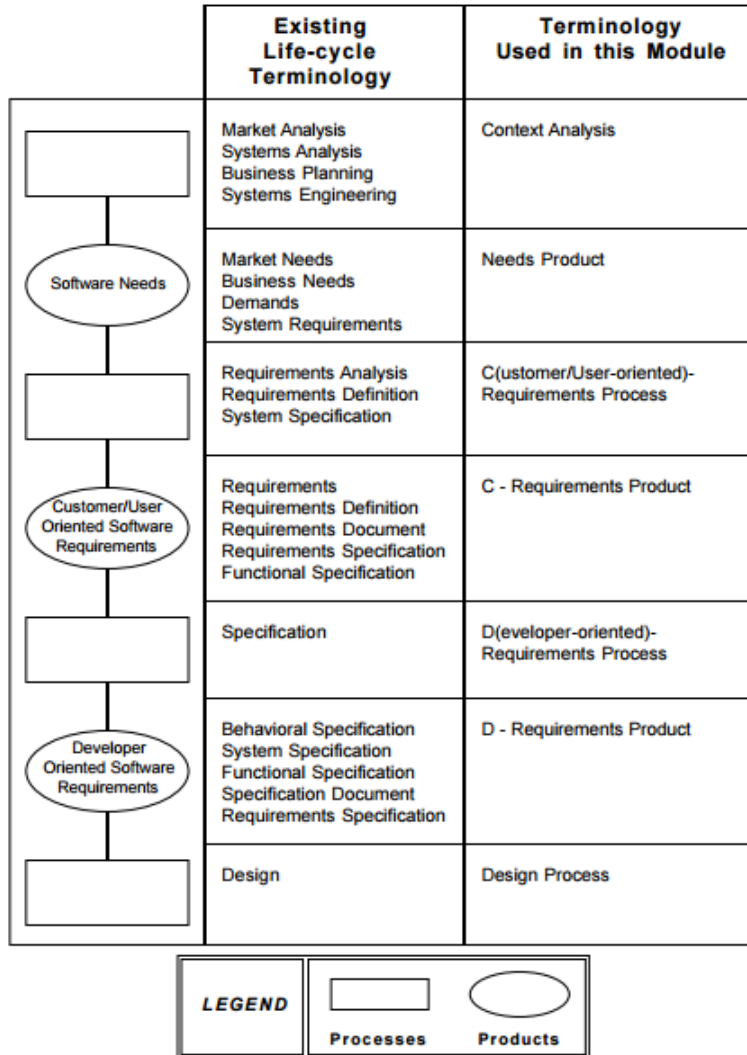
- Específica (Specific): a tarefa tem que ser específico o suficiente para que todo o grupo possa entender o que se reflete na mesma.
- Measurable (Mensurável): pode-se dizer que já está terminado? Deve ser mensuráveis e entender que a equipe tem que concordar que o processo faz o que se destina, os testes forem concluídos eo código foi programado.
- Achievable (Archiveable): o programador deve ser capaz de executar a tarefa que é feita. Há uma regra na equipe de desenvolvimento onde qualquer pessoa pode pedir ajuda quando você precisar dele, o que certamente garante que a tarefa estiver concluída.
- Relevante (relevante): cada tarefa deve ser relevante e contribuem significativamente para o desenvolvimento da forma do projecto. As histórias são divididas em tarefas para o benefício dos desenvolvedores e acabar com elas explicar e justificar para o cliente.
- Tempo box (Time - encaixotado): a tarefa é limitada a uma duração específica. Ele pode ser estimado em horas ou dias, mas se você não pode executar deve haver uma expectativa de quando pedir ajuda. Se uma tarefa é mais complicada do que o esperado, a equipe deve saber dividir a tarefa, mudar o desenvolvedor ou fazer alguma coisa para ajudar a executar a tarefa no tempo estimado.

Para Brackett (1990), elementos que são integrados no ciclo de vida terminologia mostrado na Figura 1 são dadas quando eles determinaram os requisitos gerais de sistema, tomou a decisão de que determinadas funções são realizadas por software:

- 1) O processo deve começar com os requisitos de definir e analisar o contexto desta questão, ou seja, uma breve descrição da necessidade,
- 2) Definir Necessidade: isso inclui elementos externos da empresa, tais como o mercado, a empresa necessidades internas, a demanda real e potencial, bem como os requisitos específicos de sistemas que ajudam as decisões na empresa,
- 3) Definir o processo de requisitos, ou seja, evitar o excesso de detalhe, flexibilidade devem estar presentes para que a equipe pode ajustar a quantidade da necessidade do utilizador está a ser implementado,
- 4) Identificar os requisitos do produto é para ser capaz de desenvolver em qualquer sequência e que a equipe pode usar para planejar,

- 5) Definir a exigência de processo orientado desenvolvedor,
- 6) os requisitos do produto delinear (especificação do comportamento do usuário final e do sistema) e, finalmente,
- 7) O processo de projeto envolverá analisar e testar documento critérios de aceitação ou lista para a história na definição do ciclo de vida.

Figura 1. Terminología usada para definir el ciclo de vida



Fuente: Brackett (1990).

Noções básicas de gestão e tecnologia

Por outro lado, é necessário definir o papel da gestão e da tecnologia, para aplicar a engenharia de requisitos, como tais conceitos a partir das contribuições de Valência (1996), Mora (1999), analisou bid SECAB-CINDA (1990), e Zoltan (1995).

No geral, os conceitos de gestão, gestão e gestão são sinónimos, apesar dos grandes esforços e discussões que foram feitas para diferenciar entre eles. Na prática, observa-se que o termo foi traduzido como a administração de administração, mas também como de gestão.

Nessa concepção, é para o gerente de olhar para o ambiente de modo que a organização pode gerar desenvolvimento: levar recursos e produzir mais recursos. Administrador responsável por mais manutenção e conservação, enquanto a administração é concebida funcional ou vertical (Valencia, 1996).

Mora (1999) afirmou que a gestão é linear ou tradicional, onde ela é sinônimo de gestão. Para administração é o conjunto de passos que são realizados para desenvolver um processo ou para obter um produto em particular. Supõe-se que as actividades de direcção e do governo para fazer as coisas funcionarem, com capacidade para gerar processos de transformação da realidade.

Por outro lado, o BID-SECAB-Cinda (1990) atribui o seguinte posicionamento da tecnologia de acordo com o grau de incorporação e do grau de modernidade:

- Grau de incorporação:
 - Hardware: a tecnologia da máquina built-in.
 - Software: é sem personalidade jurídica e é apresentado através de revistas, livros, manuais, vídeos, tecnologia de software de computador.
 - Orgware: estruturas organizacionais.
 - Humanware: é o built-in pessoas que têm um "know-how".
- b) Grau de modernidade:
 - Tecnologia Primitiva: corresponde ao primitivo, escravos e tempos feudais.
 - A tecnologia moderna: Em contraste com a primitiva é produzido nas últimas décadas.

- Tecnologia para trás: é que "tem sido superada em algum fator", por exemplo, a máquina de escrever elétrica com relação à escrita de memória.
- Tecnologia de ponta: é aquele que foi produzido apenas. É "fresco do forno".

Ser multi e interdisciplinar, a própria gestão da tecnologia não pode ser objecto de qualquer profissão ou disciplina (Zoltan, 1995). É um processo que lida com as interfaces entre instituições de ciência, engenharia, economia e gestão.

Gestão de tecnologia promove a organização e execução de tarefas em estreita colaboração com os agentes (investigadores, engenheiros, cientistas, tecnólogos), mas não é, em si, nem contém em si a condução da investigação, da inovação ou de controlo qualidade (ibid., 1995).

No glossário de termos BID-SECAB-Cinda (1990), indica que a gestão da tecnologia é a disciplina em que o conhecimento de engenharia, ciência e gestão são misturados, a fim de realizar o planeamento, desenvolvimento e implementação de soluções tecnológicas que contribuam para alcançar os objectivos estratégicos e técnicos de uma organização.

De acordo com Zoltan (1995), a gestão tecnológica é ao nível de um país. Uma vez que este está no mercado global de tecnologia, os clientes estrangeiros de empresas nacionais, mas também concorrência são as políticas governamentais, bem como um conjunto de regras internacionais de comércio, investimento, qualidade, meio ambiente, patentes, entre outros.

Por outro lado, no interior do país são: um governo com políticas de ciência, tecnologia e educação; um mercado de tecnologia nacional e, finalmente, as empresas com clientes e concorrentes e todo o contexto nacional e global deve tomar uma série de decisões sobre tecnologia. Eles devem decidir se a fazer o seu próprio desenvolvimento ou buscar tecnologia no exterior; no primeiro caso, eles devem tornar a inovação na segunda avaliar, seleccionar, negociar, comprar, adaptar, entre todas aquelas actividades que exigem todo um conceito de planeamento estratégico (ibid., 1995).

METODOLOGIA

Em seguida, os objectivos gerais, objectivos e específicos, bem como a hipótese e o método é apresentado para proporcionar uma visão geral para descrever o processo de investigação.

Objetivo geral

Pergunte a um processo de Engenharia de Requisitos através de Gestão e Tecnologia para a comparação de requisitos, técnicas, ferramentas e modelos para a aplicação em um serviços de informática de negócios, para satisfazer as necessidades das condições comumente omitidos necessidades de desenvolvimento.

Objetivos específicos

- Obter informações ao requisitos, técnicas, ferramentas e modelos Engenharia de Requisitos.
- Realizar características comparativas.
- Para demonstrar quais são as vantagens, desvantagens, importância e abrangência.

Hipótese

Destina-se a delinear uma engenharia de requisitos de procedimento, caracterizado pela sua importância e abrangência para o usuário, considerando-se os melhores resultados das abordagens discutidas e apoiados por vários métodos, técnicas e ferramentas já desenvolvidas. Também visa unificar a terminologia utilizada no campo da Engenharia de Requisitos, eliminando as diferenças que possam surgir como resultado da confusão semântica e melhorar o processo dele.

Portanto, a seguinte hipótese é proposto:

H_i: Se uma empresa aplica requisitos de serviços de informática de engenharia, então a eficiência será alcançada no processo de.

H_o: Se uma empresa não se aplica Requisitos serviços de informática de engenharia, então não haverá eficiência no processo.

Método

A pesquisa foi essencialmente constitui a abordagem qualitativa documental, já que a população e amostra é constituída por documentos e constitui uma estratégia onde observa-se e reflectir sistematicamente sobre as realidades utilizando para isso diferentes tipos de documentos (Gallant 2011). A população documentário é composta de referências de uso e conhecimento científico universal, basicamente, especializada artigos científicos nacionais e internacionais. Analisa, interpreta, apresenta dados e informações sobre um determinado assunto de qualquer ciência, usando análise e cuja finalidade é a obtenção de resultados que podem servir como base para o desenvolvimento da criação científica (Cortés, 2004).

Discussão e conclusões

É necessário priorizar, definir e implementar requisitos de processos de engenharia de um processo formal com atividades e produtos claramente definidos, onde um documento contendo os resultados desta etapa para o bom desenvolvimento do projeto a ser implementado também está localizado.

Quando não há nenhum documento que contém os processos formais, eles não podem ser contemplar adequadamente os custos, o tempo de desenvolvimento, riscos e estimativas que surgem durante a vida do projeto e, portanto, os requisitos não se concretizem e do produto você deseja obter falhar ou não atender às expectativas dos clientes e principal desenvolvedor do mesmo.

Ela exige requisitos de documentação e manutenção de um produto de software específico tendo em conta a contribuição de Engenharia de Requisitos.

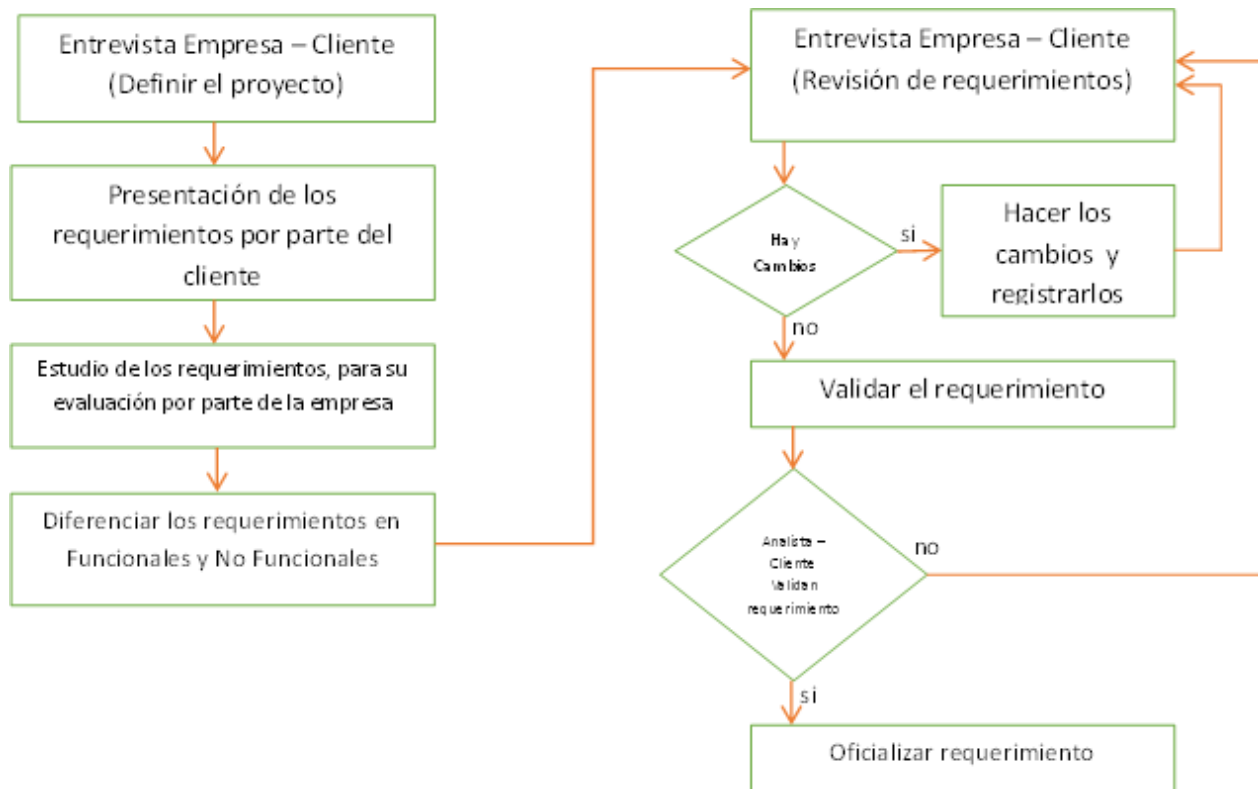
E determinar a viabilidade de realização do software através da obtenção e análise de requisitos e especificação formal.

É necessário definir o procedimento através do Gerenciamento de Requisitos e Tecnologia Engenharia como elemento fundamental para o cumprimento dos objectivos estabelecidos em

qualquer projeto para implementar na organização. A fase de validação do procedimento proposto deve ser realizada através da aplicação de um estudo de caso real aplicado à organização pela aprovação do cliente.

Uma proposta que começa com o processo de Engenharia de Requisitos para comparar os requisitos, técnicas, ferramentas e modelos de aplicação em uma empresa de serviços informáticos, como parte da análise nesta pesquisa é apresentada na figura 2 abaixo.

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 proposta gerada como resultado da análise teórica, contém o desenho de actividades comuns e o formato a ser realizada para o projeto, que irá incorporar características das metodologias de engenharia de requisitos, buscando sempre atender às necessidades de condições normalmente omitidos no desenvolvimento de requisitos.

Conclui-se que em qualquer organização deseja implementar através de requisitos de gestão de tecnologia e engenharia, devido na primeira entrevista (entre a empresa que oferece o serviço e cliente) definir o projeto, expectativas e alcance; bem como o impacto dentro da organização e externamente.

Uma vez que o projeto é exposto deve ser submetido necessidades dos clientes, que serão analisados pela sociedade oferente e determina quais são funcionais e não-funcionais.

Devemos fazer uma segunda entrevista onde os requisitos são revistos, as mudanças são diagnosticados (se houver) de cada um deles e validado. Uma vez validado por ambas as partes, estes são formalizados e começar a trabalhar sobre esse pedido.

A utilização eo desempenho de um bom modelo aplicado em Engenharia de Requisitos é usado para identificar e definir todas as atividades envolvidas nos requisitos de descoberta, documentação e manutenção para o desenvolvimento bem sucedido de um projeto e ajudar a determinar a viabilidade processo para ver se isso é viável ou não.

Ao receber as ferramentas que permitem que o projeto está bem documentado, e onde tanto o cliente quanto o desenvolvedor usar a mesma linguagem, uma especificação formalmente alcançado, a obtenção de uma validação fio verifica se realmente pedindo o cliente define o sistema que está sendo desenvolvida (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.lyftenbloggie&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.