



**Universidad de las Ciencias Informáticas
Facultad 3: Turismo y Negocio**

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas**

Título:

**GUÍA PARA LA APLICACIÓN DE TÉCNICAS
PARA EL DESARROLLO DE REQUISITOS
EN LOS PROYECTOS SOFTWARE DE LA FACULTAD No 3.**

Autora: Yehilit Gil Montes

**Tutor(es): Ing. Janet Rodríguez Febles
Consultante: Dr. Rolando Alfredo Hernández León**

**Junio 2007
“Año 49 de la Revolución”**



DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al <nombre área> de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los:

___ días del mes de _____ del año _____.



OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: <título del trabajo de diploma>

Autor: <nombres y dos apellidos del o los autores>

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

<Aquí el tutor debe expresar cualitativamente su opinión y medir (usando la escala: muy alta, alta, adecuada) entre otras las cualidades siguientes:

- Independencia
- Originalidad
- Creatividad
- Laboriosidad
- Responsabilidad

<Además, debe evaluar la calidad científico-técnica del trabajo realizado (resultados y documento) y expresar su opinión sobre el valor de los resultados obtenidos (aplicación y beneficios) >

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de <nota>. <Además, si considera que los resultados poseen valor para ser publicados, debe expresarlo también>

Firma

Fecha



DATOS DE CONTACTO

"[insertar breve curriculum e información de contacto del tutor]"

"[insertar breve curriculum e información de contacto del asesor]"

"[insertar breve curriculum e información de contacto del consultante]"



AGRADECIMIENTOS

- *Agradecimientos especiales a mis padres Marisel y Luis por ser tan lindos, estar siempre conmigo y brindarme todo su amor y apoyo.*
 - *A mi hermanita querida, Mileydis y por esos sobrinos tan bellos que me dio.*
 - *Gracias a todos mis amigos, que son pocos pero muy especiales a Yaimilita y Yaise que los quiero mucho.*
 - *A mis otros padres Cary y Alejo, por no dejar de quererme y a Jesús ¿Por qué no?*
 - *Gracias a mi primo Juancy y a su esposa Zunilda por todo su apoyo en mi carrera y que se han comportado conmigo como si fueran mis propios padres.*
 - *A mi vicedecano de Residencia y Extensión Universitaria, Pascual Verdecia Vicet, quién me ayudó mucho en la realización de mi tesis, por ser tan atento y ser un excelente amigo.*
 - *A mis tutores Rolando A. Hdez León y Janet Rodríguez Febles.*
 - *Karina, gracias por tu ayuda.*
 - *Al profesor Rolan por darme una nueva oportunidad.*
 - *A mi fiel Sivert por todos sus consejos, ayuda y sobre todo mucho amor.*
 - *A la profesora Maydelis Calderón, gracias por sus ideas y apoyo.*
 - *A mi prima Lolita por ayudarme en mi etapa final.*
 - *A nuestra gran Revolución y a la Universidad de Ciencias Informáticas por darme el derecho de estudiar, formarme profesionalmente y prepararme para la vida.*
-Y a los, que de una forma u otra contribuyeron en mi formación y aportaron su granito en el desarrollo del trabajo de diploma.....

A todos, muchas gracias.



DEDICATORIA

Especialmente a mis padres.

“La idea fundamental es que se convierta (la Informática) en la rama más productiva, aportadora de recursos para la nación”

Fidel Castro Rúz



RESUMEN

En el proceso de desarrollo de software, la aplicación de la Ingeniería de Software se utiliza para garantizar la calidad de un producto de software. Una de las etapas consideradas más importantes del proceso de desarrollo es la Ingeniería Requisitos, proceso en el cual se definen inicialmente las características y restricciones con las que debe contar el sistema en desarrollo. Para todo tipo de proyecto definir los requisitos es un punto indispensable.

Durante el desarrollo de requisitos de los proyectos de la Universidad de las Ciencias Informáticas (UCI) se ha incidido en muchas deficiencias que han provocado atrasos en los cronogramas de entrega del producto final y han afectado la calidad de los mismos. Por esta situación es que el presente trabajo realiza un estudio donde se habla sobre el proceso de desarrollo de requisitos, cuáles son las actividades y técnicas que se deben utilizar para llevar a cabo un desarrollo de requisitos con la calidad requerida. Por lo anteriormente planteado surge la necesidad de realizar una guía para la aplicación de técnicas para el desarrollo de requisitos en los proyectos de la Facultad 3 (F3) de la UCI con el objetivo de contribuir a una organización del proceso, que garantice la calidad y claridad de la información obtenida.

PALABRAS CLAVES

Guía, técnicas de aplicación, desarrollo de requisitos, proceso de desarrollo, software, Ingeniería de Software, Ingeniería de Requisitos.



TABLA DE CONTENIDO

AGRADECIMIENTOS	5
DEDICATORIA.....	6
RESUMEN.....	7
TABLA DE CONTENIDO	8
INTRODUCCIÓN.....	10
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	15
1.1 INTRODUCCIÓN	15
1.2 INGENIERÍA DE SOFTWARE	15
1.3 INGENIERÍA DE REQUISITOS.....	19
1.4 MODELOS DE PROCESO DE LA INGENIERÍA DE REQUISITOS.....	22
1.5 DESARROLLO DE REQUISITOS.....	25
1.6 REQUISITOS DENTRO DE LAS METODOLOGÍAS DE DESARROLLO	31
1.6.1 Metodología ágil	31
1.6.2 Metodología pesada (No ágil).....	33
1.7 TÉCNICAS PARA EL DESARROLLO DE REQUISITOS.....	35
1.8 CONCLUSIONES PARCIALES.....	47
CAPÍTULO 2: PROPUESTA.....	48
2.1 INTRODUCCIÓN	48
2.2 SITUACIÓN ACTUAL DE LA ORGANIZACIÓN DE LA PRODUCCION DE SOFTWARE EN LA FACULTAD # 3 EN LA UCI.	48
2.3 CARACTERIZACIÓN DEL EQUIPO DE DESARROLLO EN EL DESARROLLO DE REQUISITOS.	55
2.4 SELECCIÓN DE LAS ACTIVIDADES DEL DESARROLLO DE REQUISITOS (Extracción, Análisis, Especificación, Validación).....	57



2.5	PROPUESTA: GUÍA PARA LA APLICACIÓN DE TÉCNICAS PARA EL DESARROLLO DE REQUISITOS.....	58
2.5.1	Guía para la aplicación de técnicas para el desarrollo de requisitos.....	59
2.6	VALIDACIÓN DE LA PROPUESTA	68
2.7	CONCLUSIONES PARCIALES.....	70
	CONCLUSIONES GENERALES.....	71
	RECOMENDACIONES	72
	REFERENCIAS BIBLIOGRÁFICAS	73
	BIBLIOGRAFÍA.....	75
	GLOSARIO DE TÉRMINOS.....	77



INTRODUCCIÓN

Comienza el tercer milenio, se inicia la Batalla de Ideas que sostiene cerca de 200 programas sociales para la preparación cultural e integral y la vida del pueblo, se impulsa el uso masivo de las nuevas tecnologías de la informática y las comunicaciones (TIC) y con ellos se inicia un proceso de transformación en todos los sectores de la sociedad.

Como iniciativa del Comandante en Jefe Fidel Castro Rúz, surge la Universidad de Ciencias Informáticas (UCI) para formar talentos que desempeñen un papel importante para la producción de software y en la prestación de servicios informáticos.

La universidad, posee un gran reto ante la era del conocimiento donde existe un mercado internacional muy dinámico y competitivo, y la calidad se impone como un requisito indispensable. Es importante mencionar que el principal objetivo de la institución se fundamenta en la vinculación Formación-Docencia-Investigación. La producción como esfera importante, incrementa el nivel de compromiso con los clientes e impone el desarrollo de sistemas con calidad, eficiencia y en tiempos de poca duración. El riesgo al fracaso está inherente y se fundamenta en la poca experiencia del equipo de desarrollo en ambientes productivos, en las responsabilidades docentes presentes en todo el ciclo de desarrollo del producto, añadiendo en algunos casos la poca presencialidad de los clientes que aumenta las barreras de la comunicación existentes y dificulta el entendimiento de las necesidades que poseen.

Los requisitos definen qué es lo que el sistema debe hacer, para esto se identifican las funcionalidades requeridas y las restricciones que se imponen, es una característica que un sistema debe tener para cubrir las necesidades de los usuarios finales.



La ingeniería de requisitos facilita el mecanismo apropiado para comprender lo que quiere el cliente, analizando necesidades, confirmando su viabilidad, negociando una solución razonable, especificando una solución sin ambigüedades, validando la especificación y gestionando los requisitos para que se transformen en un sistema operacional.

Wieggers define dos componentes que recogen diferentes disciplinas de ingeniería de requisitos, ellas son el Desarrollo de Requisitos y la Gestión de Requisitos.

La Gestión de Requisitos, se refiere a un conjunto de actividades que ayudan al equipo de trabajo a identificar, controlar y seguir los requisitos y los cambios en cualquier momento. Recordemos que los requisitos del sistema cambian y el deseo de cambiarlos persiste a lo largo del ciclo de desarrollo. (PRESSMAN, 2005)

El Desarrollo de Requisitos por otra parte, se manifiesta en las actividades de extracción, análisis, especificación y validación. (Dávila, 2001)

Extracción: Esta fase representa el comienzo de cada ciclo. Se refiere a las actividades involucradas en el descubrimiento de los requisitos del sistema.

Análisis: Sobre la base de la extracción realizada previamente, comienza esta fase en la cual se apunta a descubrir problemas con los requisitos del sistema identificados hasta el momento.

Especificación: En esta fase se documentan los requisitos acordados con el cliente, en un nivel apropiado de detalle.

Validación: La validación es la etapa final de la Ingeniería de Requisitos. Su objetivo es validar los requisitos, es decir, verificar todos los requisitos que aparecen en el documento



especificado para asegurarse que representan una descripción, aceptable del sistema que se debe implementar.

En los proyectos productivos de la Universidad de las Ciencias Informáticas, existen muchas dificultades respecto a la puesta en práctica de la Ingeniería de Requisitos. Al realizar un estudio relacionadas de las dificultades presentes, se manifiesta la poca preparación del los integrantes del equipo de desarrollo para realizar las diferentes fases que lo componen. Los estudiantes responsables de estos procesos no se seleccionan de acuerdo a sus cualidades, expectativas y habilidades. El tiempo que se le dedica a estos procesos es muy poco y no se le da un tratamiento adecuado al Desarrollo de los Requisitos. Estas causas provocan una definición inadecuada del alcance del sistema que en algunas ocasiones visualizan elementos innecesarios, los usuarios en ocasiones no están seguros de sus necesidades, provocando conflictos con las necesidades de otros usuarios y requisitos ambiguos sin estabilidad. Hay que ser cuidadoso para no realizar falsos requisitos, porque los clientes son cada vez más exigentes al adquirir los productos y esto hace casi imprescindible la necesidad de llevar a cabo un desarrollo correcto de los mismos. Además, en muchas ocasiones la implementación del sistema se comienza sin una previa validación de los requisitos planteados, causas que afectan en la calidad del producto y que conllevan a retrasos y por ende incumplimientos en la fecha de entrega.

Como parte de la **situación problemática** anteriormente planteada surge el siguiente **problema de la investigación**, ¿Cómo facilitar el trabajo del analista durante la extracción, análisis, especificación y validación de los requisitos para el desarrollo de los proyectos de la facultad 3?

El **objeto de Investigación** se encuentra centrado en desarrollo de requisitos, especificando como el **campo de acción** las técnicas para su desarrollo.



El **Objetivo General** de esta tesis se fundamenta en elaborar una guía para la aplicación de técnicas de desarrollo de requisitos en los proyectos de la F3 de la UCI que permita proyectar la ejecución de ese proceso de una forma más eficiente.

Se plantea como **hipótesis** que si se desarrolla una guía para la aplicación de técnicas de desarrollo de requisitos en los proyectos de la F3 de la UCI, se podrá implementar la ejecución de ese proceso de una forma más eficiente.

Como **Objetivos específicos** se tienen:

- Diagnosticar la situación actual de la organización del proceso de producción de software en la Facultad # 3 de la UCI y el tratamiento que se le ofrece a los requisitos.
- Elaborar una guía para la aplicación de técnicas de desarrollo de requisitos en los proyectos de la F3 de la UCI.
- Validar la propuesta mediante criterios de especialistas.

El presente trabajo centra sus esfuerzos en propiciar una guía para el personal vinculado al desarrollo de requisitos, mediante diferentes técnicas que sirven de punto de partida para mejorar el entendimiento de lo que realmente necesita el sistema a construir.

Este trabajo se realizó con una estrategia de investigación de tipo descriptiva, su principal objetivo es describir el fenómeno y reflejar lo esencial y más significativo del mismo, sin tener en cuenta las causas que lo originan, para lo que es necesario captar sus relaciones internas y regularidades, así como aquellos aspectos donde se revela lo general. En este tipo de investigación es de principal importancia la profundidad teórica del planteamiento investigativo, ayuda a comprender el valor científico de los resultados obtenidos.



INTRODUCCIÓN

Para la realización de este trabajo se utilizaron como métodos científicos de investigación, los métodos teóricos lógicos específicamente el hipotético deductivo, métodos empíricos, además los particulares, los cuales son entrevistas y encuestas. Para la evaluación de los resultados de las encuestas y entrevistas se utilizó el método porcentual.



CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 INTRODUCCIÓN

En este capítulo a partir de una revisión amplia de la literatura más actualizada en esta temática, se hace un análisis de algunos conceptos y elementos que son de gran importancia para elaborar una guía para la aplicación de técnicas de desarrollo de requisitos en los proyectos de la F3 de la UCI, con el objetivo de facilitar el proceso de producción de software y la organización de los proyectos de la facultad. Se describe el papel de la Ingeniería de Requisitos en la ingeniería y desarrollo de software, su importancia en el desarrollo de los mismos y un estudio de las metodologías y técnicas para el desarrollo de los mismos.

1.2 INGENIERÍA DE SOFTWARE

El software

En computación, procedimientos y reglas lógicas escritas en la forma de programas y aplicaciones, que definen el modo de operación de la computadora. [1]

Conjunto de instrucciones escritas en un determinado lenguaje, que dirigen a un ordenador para la ejecución de una serie de operaciones, con el objetivo de resolver un problema que se ha definido previamente. [2]

Existen numerosas definiciones de software, analizando cada una, todas tienen un mismo significado desde los puntos de vista de sus autores, y según Pressman el software de computadora es el producto que diseñan y construyen los ingenieros de software. Esto abarca programas que se ejecutan dentro de una computadora de cualquier tamaño y arquitectura. [3]



El “IEEE Standard Glossary of Software Engineering Terminology” (Std. 610.12-1990) ha desarrollado una definición más completa para ingeniería del software: “(1) La aplicación de un enfoque sistemático, disciplinado y cuantificable para el desarrollo, operación y mantenimiento del software; es decir, la aplicación de ingeniería al software. (2) El estudio de enfoques en (1)”.

Es una tecnología multicapa en la que se pueden identificar los métodos, el proceso y las herramientas, ilustrada en la **Fig. 1**. Es una tecnología que indica cómo construir técnicamente un software: económico, fiable y que funcione eficientemente. [3]

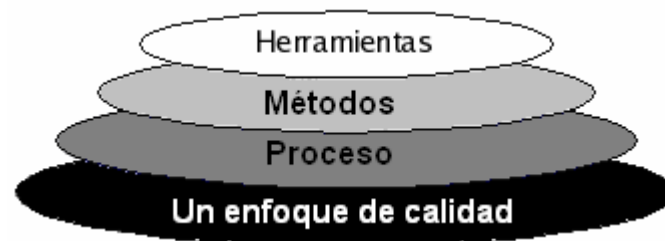


Figura 1: Capas de la Ingeniería de Software.

Tomando como base de las referencias de las definiciones anteriormente mencionadas podemos afirmar que evidentemente la ingeniería de software es un proceso completo, compuesto por métodos, herramientas y tecnologías por las que de una forma eficiente y económica transita un producto software obteniendo gran calidad del mismo.

Varios de los objetivos que persigue la Ingeniería de software es suministrar a los desarrolladores las bases para construir un producto software de calidad y facilitar el control del proceso de desarrollo de software, en este sentido de las definiciones que se encuentran en la Web, se tomó la siguiente:

Proceso

Un proceso define quién, está haciendo qué, cuándo, y cómo alcanzar un determinado objetivo. [4]



La Ingeniería de Software juega un papel fundamental, con la aplicación de sus métodos, técnicas y herramientas facilita el control del proceso de desarrollo, abasteciendo a los desarrolladores las bases para construir un proyecto de software con calidad y de una forma más eficiente.

Proceso de desarrollo de la Ingeniería de Software

Un proceso de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente. [5]

El proceso de ingeniería de software se define como un conjunto de etapas parcialmente ordenadas con la intención de lograr un objetivo, en este caso, la obtención de un producto de software de calidad.

El proceso de desarrollo de software "es aquel en que las necesidades del usuario son traducidas en requisitos de software, estos requisitos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo". Concretamente "define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un cierto objetivo" [6]

El proceso de desarrollo de software requiere por un lado un conjunto de conceptos, una metodología y un lenguaje propio. A este proceso también se le llama el ciclo de vida del software que comprende cuatro grandes fases:(Jacobson 1998)

Concepción

Define el alcance del proyecto y desarrolla un caso de negocio.

Elaboración

Define un plan del proyecto, especifica las características y fundamenta la arquitectura.



Construcción

Crea el producto.

Transición

Transfiere el producto a los usuarios.

Generalmente este proceso conlleva a una serie de pasos para su desarrollo, los cuales se especifican a continuación:

- **Análisis de requisitos**

Extraer los requisitos de un producto de software es la primera etapa para crearlo. No basta con que los clientes sepan lo que el software tiene que hacer, se requiere de habilidad y experiencia en la ingeniería de software para reconocer requisitos incompletos, ambiguos o contradictorios.

- **Especificación**

Es la tarea de describir detalladamente el software a ser escrito, en una forma matemáticamente rigurosa. En la realidad, la mayoría de las buenas especificaciones han sido escritas para entender y afinar aplicaciones que ya estaban desarrolladas. Las especificaciones son más importantes para las interfaces externas, que deben permanecer estables.

- **Diseño y arquitectura**

Se refiere a determinar ¿Cómo? funcionará el sistema de forma general sin entrar en detalles.

- **Programación**

Reducir un diseño a código, esta puede ser la parte más obvia del trabajo de ingeniería de software, pero no es necesariamente la porción más larga.

- **Prueba**

Consiste en comprobar que el software realice correctamente las tareas indicadas en la fase de especificación. Una técnica de prueba es probar por separado cada módulo del software, y luego probarlo de forma integral.



- **Documentación**

Realización del manual de usuario, y posiblemente un manual técnico con el propósito de mantenimiento futuro y ampliaciones al sistema.

- **Mantenimiento**

Mantener y mejorar el software para enfrentar errores descubiertos y nuevos requisitos. Esto puede llevar más tiempo incluso que el desarrollo inicial del software. Alrededor de 2/3 de toda la ingeniería de software tiene que ver con dar mantenimiento. Una pequeña parte de este trabajo consiste en arreglar errores. La mayor parte consiste en extender el sistema para hacer nuevas cosas. De manera similar, alrededor de 2/3 de toda la ingeniería civil, arquitectura y trabajo de construcción es dar mantenimiento. [7]

En toda la definición del proceso de desarrollo de software existen un conjunto de requisitos que se transforman en software. De esta manera, como parte del proceso de desarrollo de un sistema, el equipo de desarrollo se enfrentan a la dificultad de identificar los requisitos, que son la base fundamental del producto a desarrollar, es aquí donde la Ingeniería de Análisis de Requisitos juega un papel importante, como una de las disciplinas más significativas dentro del proceso de desarrollo de software. Esta disciplina se encarga de determinar las necesidades del cliente, y convertir la declaración de estas necesidades en una descripción completa, precisa y documentada de los requisitos porque sin una buena especificación de los mismos trae riesgos y afecta la calidad del producto.

1.3 INGENIERÍA DE REQUISITOS

¿Qué se define por requisitos?

Un requisito es un atributo necesario en un sistema, una declaración que identifica una capacidad, característica, o el factor de la calidad de un sistema en la orden para tener el valor y la utilidad a un cliente o el usuario. [8]

(1) Condición o capacidad que necesita un cliente o usuario para solucionar un problema o alcanzar un objetivo. (2) Condición o capacidad que debe poseer un producto o componente



para resolver o satisfacer un contrato, un estándar, una especificación, o un documento impositivo (3) una representación documentada de una condición o de una capacidad como en (1) o (2). [9]

Importancia de los Requisitos

Frederick P. Brooks, dice "La parte más difícil de construir un sistema es precisamente saber qué construir. Ninguna otra parte del trabajo conceptual es tan difícil como establecer los requisitos técnicos detallados, incluyendo todas las interfaces con gente, máquinas y otros sistemas. Ninguna otra parte del trabajo afecta tanto el sistema si es hecha mal. Ninguna es tan difícil de corregir más adelante... Entonces, la tarea más importante que el ingeniero de software hace para el cliente es la extracción iterativa y el refinamiento de los requisitos del producto." [22]

Los requisitos son importantes porque ellos proporcionan la base para todo el trabajo del desarrollo que sigue. No es un tema sencillo de escribir lo que el cliente dice que él quiere. Un problema fundamental en el negocio es que los requisitos son intrínsecamente dinámicos; ellos cambiarán con el tiempo como nuestra comprensión del problema que tratamos de resolver los cambios.

Las prácticas ineficaces de requisitos son un problema en toda la industria. Realizar un enfoque más disciplinado al desarrollo de requisitos y administración es necesario mejorar las tasas de éxito de proyecto.

Como un análisis fundamental se puede afirmar que los requisitos se deben descubrir antes de empezar a construir un producto, y que puede ser alguna funcionalidad o una cualidad que el producto debe tener. Un requisito existe ya sea porque el cliente quiere que ese requisito sea parte del producto final. Si no existe un correcto descubrimiento de los requisitos, no se puede diseñar o construir el producto correcto.



Clasificación de los requisitos

Los requisitos se pueden dividir en requisitos funcionales y no funcionales y estos a la vez se dividen en varias categorías:

- Los requisitos funcionales definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas.
- Los requisitos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo) mantenimiento, seguridad, portabilidad, estándares, etc.

Ingeniería de Requisitos

La Ingeniería de Requisitos se define, como un conjunto de actividades en las cuales, utilizando técnicas y herramientas, se analiza un problema y se concluye con la especificación de una solución. [10]

La confusión sobre terminología de los requisitos extiende incluso qué llamar la disciplina del conjunto. Algunos autores llaman la ingeniería de requisitos entera del dominio; otros se refieren como gestión de los requisitos. Encuentro útil para partir el dominio de la ingeniería de requisitos del software en el desarrollo de los requisitos y la gestión de los requisitos. [11]

En la **figura 2** Wieggers muestra la Ingeniería de Requisitos en dos componentes fundamentales: desarrollo de requisitos y gestión de requisitos.

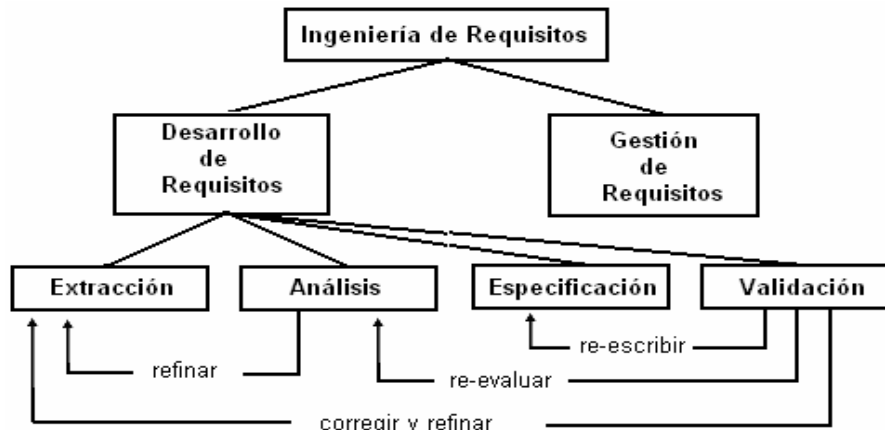


Figura 2: Subcomponentes de la ingeniería de requisitos.

Con el análisis anteriormente mencionado podemos afirmar que la Ingeniería de Requisitos es un proceso por el cual se extrae, analiza, especifica, valida y se gestiona los requisitos de un proyecto. El propósito de dicha proceso es hacer que los requisitos alcancen un estado óptimo antes de seguir adelante con el proyecto. Los buenos requisitos deben ser medibles, comprobables, sin ambigüedades o contradicciones.

1.4 MODELOS DE PROCESO DE LA INGENIERÍA DE REQUISITOS

A continuación se mostrará algunos modelos más usados y aplicables a la ingeniería de requisitos, estos son una manera de representar los procesos de una forma más simplificada.

Modelo tradicional en cascada

Este modelo describe la secuencia de las actividades, es decir que al finalizar una actividad, estos resultados dan paso al comienzo de otra actividad. Ver la **figura 3**.

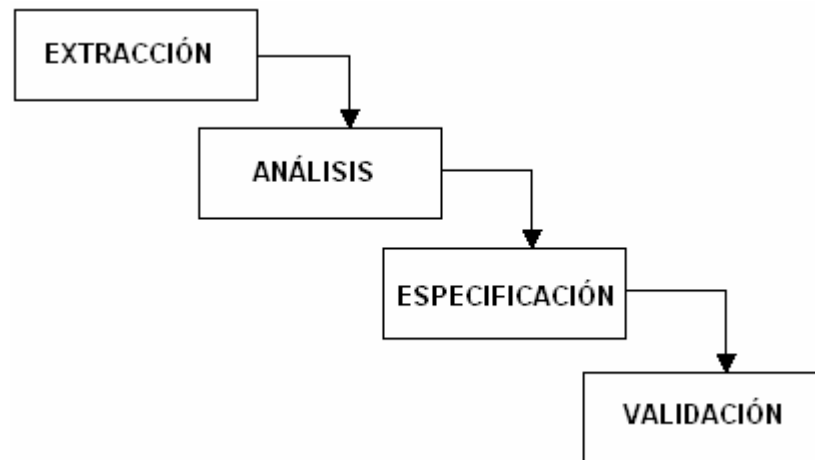


Figura 3: Modelo en cascada del proceso de Ingeniería de Requisitos

Como una descripción general y simplificada, este modelo es muy útil, sin embargo, necesitamos saber que la realidad del proceso de la Ingeniería de Requisitos es mucho más complicada que lo que se muestra en el modelo en cascada: no existen fases claramente delimitadas ya que hay una retroalimentación constante entre las distintas etapas; los requisitos del sistema van siendo modificados por motivos ajenos al proceso durante el desarrollo del mismo; se descubren problemas durante la validación que llevan a un cambio de requisitos, y todo esto hará que más de una vez se tenga que volver hacia atrás en el proceso de Ingeniería de Requisitos.

Modelo en Espiral

Es un modo alternativo de presentar modelos de actividad que toma en cuenta la retroalimentación entre etapas y la repetición de tareas, es el llamado Modelo en Espiral. **(Ver figura 4)** [12].

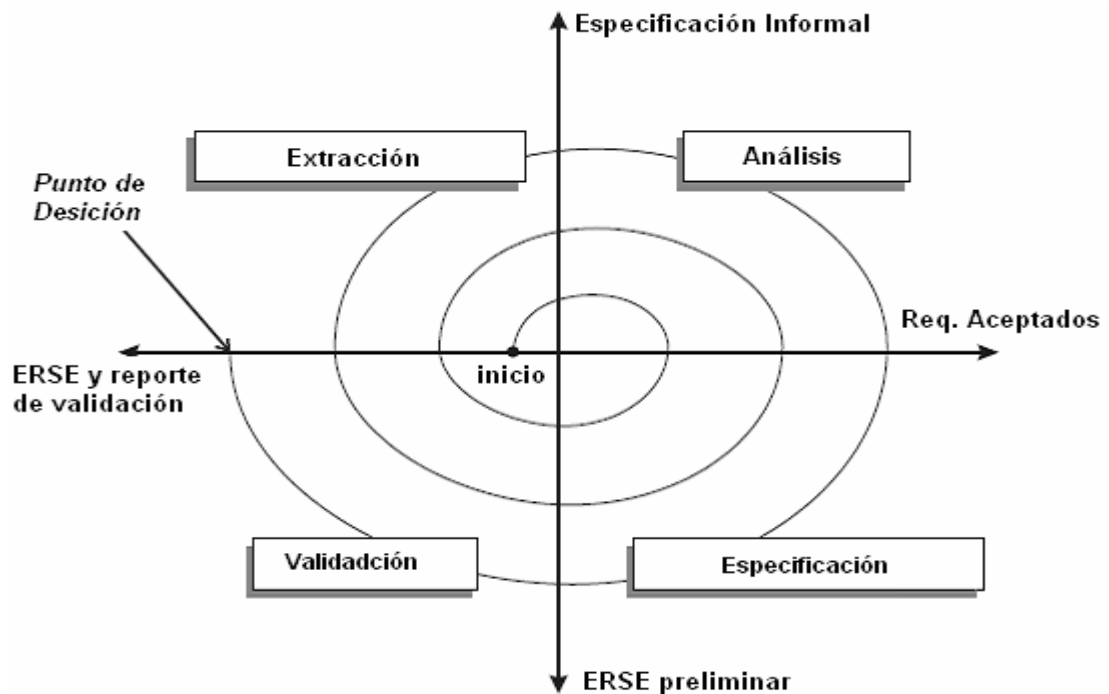


Figura 4: Modelo Espiral del Proceso de la Ingeniería de Requisitos

En este diagrama, el uso de la espiral implica que las diferentes actividades de la ingeniería de requisitos son repetidas hasta que se toma la decisión final, que es la aceptación del documento de especificación de requisitos.

Es decir, si en el diseño preliminar se encuentran problemas, entonces se recorre el ciclo nuevamente (extracción-análisis-especificación-validación) hasta que todos sean resueltos, que es lo mismo que decir que este ciclo continuará hasta que se pueda elaborar un documento aceptable.

Existen grandes diferencias en la terminología usada por los autores de la bibliografía consultada a los efectos del entendimiento de la Ingeniería de Requisitos. Es así que hay autores que ven al Análisis de Requisitos como el proceso completo de definición de requisitos y no como una etapa metodológica de la Ingeniería de Requisitos. De igual manera, la Especificación de Requisitos tiene diferentes acepciones: algunos autores se



refieren a ella como una etapa en la que se describen los requisitos y otros como a la actividad completa, desde la extracción hasta la especificación propiamente dicha.

1.5 DESARROLLO DE REQUISITOS

El proceso de desarrollo de requisitos es un conjunto estructurado de actividades, mediante las cuales se obtiene, se valida y se mantiene el documento de especificación de requisitos. Dicho proceso transita por varias actividades, estas incluyen la extracción de requisitos, el análisis, especificación y la validación de los mismos, según lo ilustrado la **figura 2**. (Wieggers, 2006)

Actividades del Desarrollo de Requisitos

Wieggers plantea que:

La *extracción*: implica las acciones por las que vamos a entender a nuestros usuarios y a descubrir sus necesidades.

Análisis de requisitos: Una actividad muy importante, que no es más que derivar requisitos más detallados de requisitos de alto nivel. El análisis también implica el crear vistas múltiples de los requisitos, tales como prototipos de modelos gráficos del análisis, y de las pruebas. Otros aspectos del análisis de requisitos incluyen prioridades de negociación, buscando para requisitos que falta, viabilidad técnica, riesgo, y los modos de fallo de evaluación. El análisis proporciona un lazo de regeneración que refine la comprensión que el analista desarrolló durante una actividad de extracción.

La especificación: implica el registrar de los varios tipos de información de los requisitos en las formas que facilitarán la comunicación entre los tenedores de apuestas del proyecto. Estas formas son tradicionalmente documentos que contienen el texto de la lengua natural. La especificación podía consistir en la información de los requisitos almacenada en una



base de datos, como en una herramienta de gerencia comercial de los requisitos, más bien que ser un documento tradicional.

La validación: se asegura de que esos requisitos estén correctos, satisfagan necesidades de cliente, y tengan todas las características de los requisitos de alta calidad. La validación puede conducir al analista a reescribir especificaciones de algunos requisitos, a valorar de nuevo el análisis inicial, o a corregir y a refinar el sistema de requisitos documentados.

La iteración es la llave al éxito en el desarrollo de los requisitos. No se debe tomar un solo paso con la extracción, el análisis, la especificación, y la validación. Se tiene que planear en ciclos múltiples, refinando progresivamente los requisitos a un nivel conveniente del detalle. Al trabajar como analista, puede ser que se hable con algunos usuarios para conseguir una cierta información, analizarla y chequearla con los mismos para clarificar cualquier cosa que no se entienda. Luego el analista prepara lo aprendido y pasa ese fragmento de una especificación a los representantes del usuario para una revisión rápida, informal. Este proceso da lugar a los ciclos múltiples de extracción interpolados con los ciclos rápidos de la revisión. Este proceso iterativo es una manera de acumular una información más de alta calidad, de filtrar hacia fuera malentendidos, y de agregar lo completo a la capa a la vez. El analista podría realizar la extracción, el análisis, la especificación, y actividades de la validación concurrentemente en diversos sistemas de requisitos.

En el proceso de desarrollo de un sistema, el equipo de desarrollo se enfrenta al problema de la identificación de requisitos. La definición de las necesidades del sistema es un proceso complejo, en él hay que identificar los requisitos que el sistema debe cumplir para satisfacer las necesidades de los usuarios finales y de los clientes. [13]

El proceso de especificación de requisitos se puede dividir en tres grandes actividades:

Captura de requisitos: La captura de requisitos es la actividad mediante la que el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible,



CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

las necesidades que debe cubrir dicho sistema. El proceso de captura de requisitos puede resultar complejo, principalmente si el entorno de trabajo es desconocido para el equipo de analistas, y depende mucho de las personas que participen en él. Por la complejidad que todo esto puede implicar, la ingeniería de requisitos ha trabajado desde hace años en desarrollar técnicas que permitan hacer este proceso de una forma más eficiente y precisa.

Definición de requisitos: analizan y definen realmente el sistema que el usuario necesita o el cliente desea.

Validación de requisitos: Los requisitos una vez definidos necesitan ser validados. La validación de requisitos tiene como misión demostrar que la definición de los requisitos define realmente el sistema que el usuario necesita o el cliente desea. Es necesario asegurar que el análisis realizado y los resultados obtenidos de la etapa de definición de requisitos son correctos.

López define un conjunto de actividades y documentos involucrados en la Ingeniería de Requisitos, las cuales se muestran a continuación: [14]

Establecimiento de la necesidad del sistema: En esta actividad se debe cumplir que el proyecto es apoyado por la gerencia de la empresa, es el momento apropiado para el compromiso y la adquisición de recursos, apoyo a la empresa a moverse hacia sus metas y dejar bien claro la necesidad del sistema.

Factibilidad del sistema: Posibilidad de realización del software con el hardware y software existentes. Además, se realiza una estimación del costo y se contrasta con los beneficios que el software brindará a la empresa. Esta actividad no debe realizarse hasta asegurar que se establecieron las necesidades del sistema.



CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Análisis de requisitos. Este es un proceso de derivación de requisitos del sistema de software a través de la observación de sistemas existentes, discusión con usuarios, análisis de tareas y así sucesivamente. Esto puede involucrar el desarrollo de uno o más modelos diferentes del sistema, que ayudan al analista a entender el sistema que será especificado. También pueden desarrollarse prototipos para ayudar a entender requisitos.

Definición de requisitos. Es la traducción de información recolectada durante la actividad de análisis a un documento que define el conjunto de requisitos. Éste podría reflejar con precisión qué es lo que el cliente desea. Debe escribirse de forma que sea entendible por usuarios finales y cliente del sistema.

Especificación de requisitos. En esta actividad se detalla el documento de Especificación de Requisitos de Software (ERS). Conjuntamente se puede realizar un diseño de alto nivel para ayudarse a descubrir errores en definición de requisitos, mismos que deben ser corregidos. Las especificaciones forman un puente entre requisitos y diseño. Sus entradas son el documento de requisitos y la información de los usuarios y desarrolladores. La especificación de un sistema de software describe aspectos funcionales y de calidad.

Cada actividad descrita produce un documento que pasará a ser parte del documento final de Especificación de Requisitos de Software (ERS), el cual es un documento completo que especifica qué va a realizar un sistema y no cómo.

Y por ultimo análisis sobre los planteamientos de las actividades dentro de la Ingeniería de Requisitos, Dávila plantea 4 actividades fundamentales: [15]

Extracción: Esta fase representa el comienzo de cada ciclo. Extracción es el nombre comúnmente dado a las actividades involucradas en el descubrimiento de los requisitos del sistema. Aquí, los analistas 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.

Descubrir los requisitos del sistema no sólo implica preguntar a las personas qué quieren: es un proceso delicado que involucra comprender el dominio de aplicación, es decir, obtener un conocimiento del área general de aplicación del sistema; comprender el problema en sí, lo que implica que se debe extender y especializar el conocimiento sobre el dominio general para que se aplique al cliente en particular; comprender el negocio, por tanto, se debe entender en profundidad cómo es que este sistema interactuará afectará a las partes del negocio que estarán involucradas y cómo puede contribuir a lograr las metas de la empresa; finalmente, comprender las necesidades y restricciones de los usuarios del sistema, en particular, se deben entender los procesos de trabajo que se supone que el sistema apoyará y el rol de cualquier otro sistema que actualmente se involucre en dichos procesos.

Análisis: Sobre la base de la extracción realizada previamente, comienza esta fase que se presenta sumamente compleja en un proyecto donde el dominio es desconocido en la cual sea apunta a descubrir problemas con los requisitos del sistema identificados hasta el momento.

Usualmente se hace un análisis luego de haber producido un bosquejo inicial del documento de requisitos; aquí se leen los requisitos, se conceptúan, se investigan, se intercambian ideas con el resto del equipo, se resaltan los problemas, se buscan alternativas y soluciones, y luego se van fijando reuniones con el cliente para discutir los requisitos.

Especificación: En esta fase se documentan los requisitos acordados con el cliente, en un nivel apropiado de detalle. En la práctica, esta etapa se va realizando conjuntamente con el análisis, pero podríamos decir que la Especificación es el "pasar en limpio" el análisis realizado previamente aplicando técnicas y/o estándares de documentación.



Validación: La validación es la etapa final de la Ingeniería de Requisitos. Su objetivo es verificar todos los requisitos 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 requisitos sean consistentes y que estén completos.

La validación representa un punto de control interno y externo; interno, porque se debe verificar internamente lo que se está haciendo, y externo, porque se debe validar con el cliente.

Preferentemente, el documento de requisitos obtenido en la etapa anterior sólo debería incluir los requisitos que son aceptables para los usuarios. Pero es inevitable que durante la validación se descubran algunos problemas relacionados con los usuarios, y esto se debe corregir antes de aprobarse el documento final de requisitos.

En definitiva, la validación de especificaciones realmente significa asegurarse de que el documento de requisitos represente una descripción clara del sistema, y es una verificación final de que los requisitos cubren las necesidades de los usuarios.

Documento de requisitos

El documento de requisitos describe todo lo referente a servicios y funciones que el sistema deberá suministrar, las restricciones bajo las cuáles el sistema deberá operar, las propiedades generales del sistema, las definiciones de otros sistemas con los cuáles el sistema deberá integrarse, toda la información sobre el dominio de la aplicación, las restricciones en el proceso de desarrollo y las restricciones del equipo (hardware) en que correrá el sistema, entre otros elementos. [12]

El documento de requisitos debe incluir una descripción general del sistema, glosario, definición de los requisitos funcionales y no funcionales. Para la elaboración del documento de requisitos, se puede utilizar los siguientes aspectos que se mencionan:

- Usar lenguaje simple, consistente y conciso.



- Completar el lenguaje natural con otras descripciones de requisitos.
- Especificar requerimientos cuantitativamente.
- Definir plantillas estandarizadas para describir requisitos.

1.6 REQUISITOS DENTRO DE LAS METODOLOGÍAS DE DESARROLLO

Internacionalmente existen muchas metodologías para el desarrollo de software las cuáles definen etapas y fases que incluyen de alguna manera las actividades de la Ingeniería de Requisitos. Cada una de estas metodologías tiene su propia forma de realizar este proceso de manera que se obtenga un producto con calidad y con un costo mínimo. Dentro de las metodologías más desarrolladas e importantes en el proceso de desarrollo de software a nivel internacional se encuentran: Rational Unified Process (RUP), Extreme Programming (XP). Es de interés consultar cómo aplican las actividades de la Ingeniería de Requisitos cada una de estas metodologías.

Las metodologías ágiles están revolucionando la manera de producir software, y a la vez generando un amplio debate entre sus seguidores y quienes por desconfianza o conformidad no las ven como alternativa para las metodologías tradicionales.

1.6.1 Metodología ágil

XP:(Programación Extrema)

XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. [16]



La captura de requisitos de XP: gira entorno a una lista de características que el cliente desea que existan en el producto final. Cada una de estas características recibe el nombre de historias de usuarios y su definición consta de dos fases:

Definición de las características y la dificultad de cada una de ellas: El cliente describe con sus propias palabras las características que desea que posea el software y el responsable del equipo de desarrollo le informa de la dificultad técnica de cada una de ellas y por lo tanto de su coste. A través del diálogo resultante el cliente deja por escrito un conjunto de historias y las ordena en función de la prioridad que tienen para él. En este momento ya es posible definir unos hitos y unas fechas aproximadas para ellos.

Analizar las primeras historias que serán implementadas: Se analizan las primeras historias que serán implementadas (primera iteración) y se dividen en las tareas necesarias para llevarlas a cabo. El cliente también participa, pero hay más peso del equipo de desarrollo, que dará como resultado una planificación más exacta. En cada iteración se repetirá esta segunda fase para las historias planificadas para ella.

Este proceso es una de las principales diferencias con las metodologías tradicionales. Aunque las historias de usuarios guardan cierta relación con otras técnicas como los casos de uso de UML, su proceso de creación es muy diferente. En lo que al cliente se refiere no se le exige que especifique exactamente lo que quiere al principio con un documento de requisitos de usuario. En este caso la diferencia está en que es el cliente el que tiene que escribir lo que quiere, no se permite que alguien del equipo de desarrolladores lo escriba por él.

Las historias de los usuarios se plasmarán en tarjetas, lo que facilitará que el cliente pueda especificar la importancia relativa entre las diferentes historias de usuario, así como la tarea de los desarrolladores que podrán catalogarlas convenientemente. El formato de tarjeta además es muy provechoso a la hora de realizar pruebas de aceptación.



Otras metodologías ágiles que pueden ser analizadas se tienen: SCRUM [17], Dynamic Systems Development Method (DSDM) [18], Adaptive Software Development (ASD) [19], Feature -Driven Development (FDD) [20].

1.6.2 Metodología pesada (No ágil)

RUP: (Rational Unified Process)

El Proceso Unificado es un proceso de desarrollo de software, conjunto de actividades para transformar los requisitos de un usuario en un sistema software. Sin embargo, el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto.

Dentro de los flujos de trabajo que propone RUP se encuentra la captura de requisitos. El propósito fundamental del flujo de trabajo de los requisitos es guiar el desarrollo hacia un sistema correcto. Esto se consigue mediante una descripción de los requisitos del sistema, es decir, las condiciones o capacidades que el sistema debe cumplir suficientemente buena como para que pueda llegarse a un acuerdo entre el cliente (incluyendo a los usuarios) y los desarrolladores sobre qué debe y qué no debe hacer el sistema.

La metodología RUP define la Ingeniería de Requisitos en varias fases que son:

Análisis del problema: El objetivo es entender el problema así como las necesidades de los involucrados y proponer una solución a un alto nivel. Se debe de identificar los límites de la solución y las restricciones de la solución.

Comprender las necesidades de los clientes: Se debe determinar cuáles son las mejores fuentes de información, tener acceso a ellas y determinar cuál será la mejor forma de



obtener información de ellas. Las actividades para la obtención de información es utilizar técnicas tales como: entrevistas, tormentas de ideas, prototipos conceptuales, cuestionarios.

Definición del sistema: El objetivo es traducir las necesidades de los clientes a una descripción significativa del sistema que será construido.

Analizar el alcance del proyecto: Para llevar eficientemente un proyecto es necesario asignar prioridades a los requisitos, dado que el objetivo es enfocarse en tareas que mitiguen el riesgo del proyecto o estabilizar la arquitectura de la aplicación. Requiere administrar las salidas del proyecto en sus diferentes fases.

Refinar la Definición del Sistema: El objetivo es detallar la definición del sistema para que esta pueda ser entendida por los clientes. El otro aspecto es definir como será probado el sistema.

Administrar el cambio de los Requisitos: Incluye las actividades de establecer una línea base, establecer que dependencias son importantes de dar seguimiento, establecer un seguimiento entre elementos relacionados y controlar el cambio.

La intención de estas 6 actividades es garantizar que el proceso de requisitos sea llevado de manera uniforme, garantizando que se logrará obtener todos los requisitos y que estos se encuentran en concordancia con lo deseado por el cliente. Además de garantizar que en el caso de que estos deban cambiar, el cambio será de una manera controlada (IEEE).

Para capturar los requisitos de manera eficaz, los analistas necesitan un conjunto de técnicas que les ayude a obtener una visión suficientemente buena del sistema para avanzar en los flujos de trabajo siguientes.



1.7 TÉCNICAS PARA EL DESARROLLO DE REQUISITOS.

En este punto se describe las técnicas y herramientas que se utilizan para llevar a cabo cada una de las actividades del proceso de la Ingeniería de Requisitos, se mencionan a continuación las más utilizadas: (Dávila, 2001)

- Entrevistas y cuestionarios
- Sistemas Existentes
- Brainstorming (tormenta de ideas)
- JAD (Desarrollo Conjunto de Aplicaciones)
- Arqueología de documentos
- Aprendiz
- Observación
- Run Use Case WorkShop (Talleres de Trabajo basados en los Casos de Uso)
- Prototipo Bosquejado
- Prototipo Tangible/usable
- Glosario
- ESRE (Documento de Especificación de Requisitos)
- Casos de uso
- Checklist (Lista de Chequeo)
- Grabaciones de video y audio.
- Matriz DAFO.
- Modelo de la clase Conceptual.
- Diagrama espina de pescado.
- DCO (Documento de Concepto de Operaciones)
- Diagrama de Actividad.
- Casa de calidad o QFD



Cada una de estas técnicas se aplican para efectuar un buen desarrollo de requisitos, a continuación se fundamenta cada una de ellas.

Entrevistas y cuestionarios:

Las entrevistas y cuestionarios se emplean para reunir información proveniente de personas o grupos de ellas, información que se obtiene conversando con el encuestado.

Las entrevistas se emplean para recolectar información en forma verbal, a través de preguntas elaboradas por el analista, estas preguntas se les realiza a quienes se encuentren afectado por la aplicación, a usuarios con gran nivel de conocimiento del sistema o a personas que pueden proporcionar datos de interés.

Para realizar las entrevistas, conviene llevar preparado un cuestionario. Un **cuestionario** consiste en un conjunto de preguntas presentadas a una persona para su respuesta. La forma de la pregunta puede influir en las respuestas, por lo que hay que planearlas cuidadosamente. Los cuestionarios pueden ser abiertos o cerrados.

Abiertos: Al igual, que las entrevistas, los cuestionarios pueden ser abiertos y se aplican cuando se quieren conocer informaciones generales; también son útiles al explorar el problema básico.

Cerrados: El cuestionario cerrado limita las respuestas posibles del interrogado. Este formato es el mejor método para obtener información sobre los hechos. También fuerza a los individuos para que tomen una posición y forma de opinión sobre los aspectos importantes.

Sistemas Existentes

Esta técnica consiste en analizar distintos sistemas ya desarrollados que estén relacionados con el sistema a ser construido.

Por un lado, podemos analizar las interfases de usuario, observando el tipo de información que se maneja y cómo es manejada. Esto puede ser útil para descubrir información



importante a tener en cuenta, información que tal vez el cliente/usuario haya fallado en comunicar.

Es recomendable que luego de haber analizado el sistema, se lo mostremos al cliente/usuario, ya que por su experiencia puede sugerir importantes ideas nuevas.

Brainstorming (Tormenta de ideas)

Este es un modelo que se usa para generar ideas. La intención en su aplicación es la de generar la máxima cantidad posible de requisitos para el sistema.

No hay que detenerse en pensar si la idea es o no del todo utilizable. La intención de este ejercicio es generar, en una primera instancia, muchas ideas. Luego, se irán eliminando en base a distintos criterios como, por ejemplo, "caro", "impracticable", "imposible", etc.

En esta actividad pueden participar todas aquellas personas que se encuentren afectados por la aplicación (clientes), es decir los que la utilizarán y el personal experimentado del equipo de desarrollo, como el líder de proyecto, analistas de sistema, diseñadores, etc.

Por más locas o salvajes que parezcan algunas ideas, no se las debe descartar, porque luego de maduras probablemente se tornen en un requisito sumamente útil.

A veces ocurre que una idea resulta en otra idea, y otras veces podemos relacionar varias ideas para generar una nueva.

JAD (Desarrollo Conjunto de Aplicaciones)

Es una alternativa a las entrevistas. Es una práctica de grupo que se desarrolla durante varios días y en la que participan analistas, usuarios, administradores del sistema y clientes. Tras una fase de preparación del JAD al caso concreto, el equipo de trabajo se reúne en varias sesiones. En cada una de ellas se establecen los requisitos de alto nivel a trabajar, el ámbito del problema y la documentación.

Durante la sesión se discute en grupo sobre estos temas, llegándose a una serie de conclusiones que se documentan. En cada sesión se van concretando más las necesidades del sistema. Esta técnica presenta una serie de ventajas frente a las entrevistas



tradicionales, ya que ahorra tiempo al evitar que las opiniones de los clientes se tengan que contrastar por separado, pero requiere un grupo de participantes bien integrados y organizados.

JAD tiene dos grandes pasos:

JAD/Plan cuyo objetivo es analizar y especificar requisitos.

JAD/Diseño, en el que se aborda el diseño del software.

En comparación con las entrevistas individuales JAD presenta varias ventajas por ejemplo, ahorra tiempo al evitar que las opiniones de los clientes se contrasten por separado, todo el grupo, incluyendo los clientes y los futuros usuarios, revisa la documentación generada, no sólo los ingenieros de requisitos y además implica más a los clientes y usuarios en el desarrollo.

Arqueología de documentos

Con la aplicación de esta técnica se trata de determinar posibles requisitos sobre la base de inspeccionar la documentación utilizada por la empresa; por ejemplo, boletas, facturas, remitos, etc. Esta herramienta sirve más que nada como complemento de las demás técnicas, y nos ayuda a obtener información que de otra manera sería sumamente difícil conseguir.

Se debe recolectar cualquier formulario o documento que sea utilizado para registrar o enviar información.

Aprendiz

Esta técnica se basa en la idea del maestro y el aprendiz, y es una buena forma de observar el trabajo real. Aquí, el aprendiz es representado por el analista, y el usuario/cliente cumple el rol de maestro.

El aprendiz se sienta con el maestro a aprender por medio de la observación, haciendo preguntas como ¿por qué hizo eso? y ¿qué significa eso?, y también realizando algún trabajo bajo la supervisión del maestro.



La aplicación de esta técnica es muy útil, ya que a veces es difícil para el cliente/usuario el explicar cómo realiza su trabajo. Es también una técnica apropiada para un proyecto donde el problema no es estructurado, ya que es una de las mejores formas de obtener el conocimiento que se encuentra en la "cabeza" del cliente.

Una de las posibles objeciones que se le pueden hacer a esta técnica es que su implementación requiere de mucho tiempo.

Observación

Observar como se hacen las cosas es una buena manera de entender lo que estas requieren. Conectarse íntimamente con la cultura de la organización, vivirla, es una herramienta que debe ser tomada en cuenta. Siempre tenemos que estar atentos a lo que sucede en el entorno de la organización; por ejemplo, ver cómo resuelven un problema que surge. Dentro de la estrategia de observar tenemos que tratar de buscar estructuras y patrones. La estructura del trabajo para los usuarios suele ser invisible, por lo que será nuestro trabajo realizar las abstracciones necesarias.

Run Use Case WorkShop (Talleres de Trabajo basados en los Casos de Uso)

Estos talleres de trabajo se realizan entre el cliente/usuario y el equipo de requisitos. La primera parte del taller de trabajo consiste en generar los escenarios. Para esto se necesita la información que tiene para brindar el usuario/cliente. La idea es conversar por medio de los casos de uso y extraer de los usuarios las cosas esenciales que suceden cuando ocurre un evento determinado. Así, tratamos de definir la serie de usuarios y reconocer los pasos que se realizan para el caso de uso en estudio. Luego preguntamos si los pasos registrados están bien o si hay que cambiarlos o mejorarlos. Como resultado de este proceso obtenemos un excelente bosquejo del caso de uso. Una vez finalizada la etapa anterior, el equipo de requisitos retorna a la oficina a especificar y deducir los requisitos, a partir del conocimiento previamente adquirido.



Prototipo Bosquejado

Para validar los requisitos 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 en base a los requisitos recolectados le permite al usuario realizar su trabajo de manera eficiente y efectiva.

Una clasificación de prototipos es el prototipo bosquejado. Para realizar esta clase de prototipo nos apoyamos, por un lado, en el rol del analista de requisitos, simulando las respuestas del sistema y realizando bosquejos de las interfases de usuario; y, por otro, el usuario, que es quien realiza las entradas ("utiliza el prototipo"). También podemos llevar el caso de uso y bosquejar la interfase de usuario y, mediante el diálogo, manejamos la interactividad entre el usuario y el sistema.

Prototipo Tangible/usable

Otra clasificación de prototipo se tiene el prototipo tangible/usable.

Los términos tangible y usable se refieren a desarrollar una aplicación (software) que el usuario pueda utilizar, es decir, con la cual pueda interactuar como si fuera la aplicación final. Cualquiera sea la herramienta de software, que elijamos utilizar, para desarrollar el prototipo, debemos recordar los siguientes puntos:

- Debe demandar poco esfuerzo para realizar los cambios.
- Debe poseer amplia flexibilidad para el manejo de las interfases de usuario.
- Debe consumir poco tiempo para generar un nuevo prototipo (maqueta).

Entre las desventajas más importantes de los prototipos que podemos mencionar, se encuentran:

- Costo de entrenamiento/capacitación en la herramienta.
- Costo de realizar el prototipo.
- Problema de calendario.



- Incompletitud (puede confundir a los usuarios, haciéndolos pensar que el producto final quedará como el prototipo, incompleto).

Glosario

El glosario es una simple lista de términos en donde se explica su significado. En esta lista se incluyen y definen todos los términos que requieren explicación, mejorando así la comunicación intergrupala y la comunicación con el cliente, y mitigando el riesgo de malos entendidos. Los términos que se incluyen provienen de todas las áreas del proyecto: casos de uso, terminología propia del negocio, etc.

El glosario se va actualizando durante el transcurso del proceso de Ingeniería de Requisitos, perfeccionándolo en cada nuevo ciclo.

La forma de confeccionar y representar el glosario es de la siguiente manera: **(Tabla 1)**

Término	Descripción

Tabla 1: Formato del glosario de términos

ESRE (Documento de Especificación de Requisitos)

El objetivo del documento ESRE es el de especificar los requisitos del sistema, o sea "qué" debe hacer el sistema. Solamente se incluyen los requisitos del producto. Estos requisitos los podemos clasificar en dos grandes categorías, los no-funcionales y los funcionales. En este documento debemos colocar la lista de requisitos con las respectivas referencias a los documentos de todos los casos de uso que satisfacen los requisitos.

La lista de requisitos forma parte del documento de especificación de requisitos (ESRE). En este documento se listan los requisitos funcionales que el sistema debe satisfacer.



Casos de uso

El caso de uso es un documento narrativo que describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso. Es una técnica diseñada para especificar el comportamiento de un sistema.

Este documento describe la posible secuencia de interacciones entre el sistema y uno o más actores, en respuesta a un estímulo inicial proveniente de un actor.

Los requisitos se pueden expresar de diferentes formas, desde texto sin formato estricto hasta expresiones en un lenguaje formal, pasando por todas las formas intermedias. La mayoría de los requisitos funcionales, sino todos, se pueden expresar con casos de uso.

Un caso de uso típico debe incluir:

Nombre del caso de uso

- Actores (quiénes intervienen en el caso de uso).
- Descripción del objetivo del caso de uso (Esta debe expresar lo que ocurre desde el punto de vista del usuario).
- Referencia a los requisitos específicos del sistema.
- Interfaz de usuario (IU).
- Descripción del caso de uso.

Checklist (Lista de Chequeo)

Esta herramienta es muy fácil de utilizar y proporciona una gran utilidad. En general es una lista de preguntas que el analista debe usar para evaluar cada requisito. Los analistas deben verificar y marcar los puntos de esta lista mientras leen el documento de requisitos. Cuando se descubren problemas potenciales, deben ser anotados, ya sea en los márgenes del documento, ya sea en una lista de análisis.

Las listas de chequeo son útiles porque brindan un recordatorio de lo que se debe buscar y reducen las oportunidades de pasar por alto alguna verificación importante. Y no sólo son



útiles para verificar requisitos, también se puede aplicar con los casos de uso y con los puntos a ser tratados en el plan de agenda.

Grabaciones de video y audio

Básicamente existen dos formas de utilizar las grabaciones: como registro y apoyo de las entrevistas, y para analizar algún proceso en particular.

En cuanto a su función de apoyo, es importante por cuanto permite centrar la atención en la entrevista en sí en vez de distraerse tomando notas de todo lo que se dice. Cuando se está grabando la conversación, basta con "puntear" en una libreta los temas tratados para después tener una guía básica de los temas tratados y saber en qué lugar de la grabación buscar. Además, permite analizar los temas con más detenimiento y con una visión más global, pues ya se ha conversado sobre todos los puntos necesarios y se han visto los procesos.

Cuando se trata de analizar algún proceso en particular, su ayuda es inestimable (sobre todo las filmaciones de video) ya que permite ver y analizar en detalle ese proceso la cantidad de veces que sea necesario. Y no olvidemos que filmando el lugar de trabajo estamos capturando el proceso de trabajo, lo que evita que impongamos nuestras expectativas y preferencias.

Matriz DAFO (Debilidades, Amenazas, Fortalezas, Oportunidades).

Con este análisis se intentan identificar las principales fortalezas, oportunidades, debilidades y amenazas con las que se enfrenta una empresa.

Entonces, por un lado, tenemos las oportunidades y las amenazas, que se refieren a los factores externos que pueden afectar el futuro del negocio.

Por otro lado, se encuentran las fortalezas y debilidades que son factores internos; estas fortalezas señalan ciertas estrategias cuya aplicación podría conducir al éxito, mientras que las debilidades señalan aquello que la empresa debe corregir. Observar la **figura 5**, forma de representación.

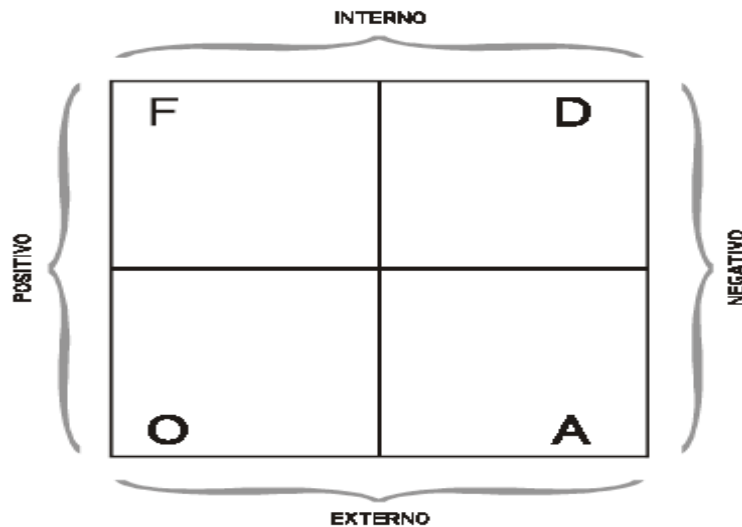


Figura 5: Matriz DAFO

Esta técnica es sumamente útil para analizar la situación de una empresa y ver de qué forma podemos ayudar a disminuir las debilidades y amenazas, y cómo podemos aprovechar las oportunidades o cómo podemos crear nuevas oportunidades y cómo hacer aún más fuertes las fortalezas. También nos es útil para analizar el impacto de la solución planteada en cada uno de los cuadros.

Modelo de la clase Conceptual.

Un modelo conceptual es una representación de conceptos del dominio del problema. Permite mostrar conceptos, asociaciones entre conceptos y atributos de conceptos. La creación del modelo también ayuda a comprender la terminología del dominio y comunica cuáles son los términos importantes y las relaciones existentes entre ellos.

Concepto: categoría de idea o cosas. La intención del concepto es la descripción de sus atributos, operaciones y significado. [21]

Esta técnica puede ser utilizada para capturar el vocabulario del sistema que se está desarrollando. Mediante ella podemos incluir abstracciones que forman parte del dominio.



Para especificar el modelo conceptual utilizamos el diagrama de clases en el cual mostramos las relaciones existentes entre las clases. Es decir, mostramos los conceptos que se manejan en el negocio y como se relacionan entre sí.

Es una buena forma para obtener un idea general de cómo funciona el negocio y la captura del vocabulario y los conceptos. También es de ayuda para incluir nuevos conceptos al glosario.

Diagrama espina de pescado.

Es una antigua pero útil técnica que sirve, en el proceso de Ingeniería de Requisitos, para analizar problemas y comprender cuáles son sus causas. El diagrama espina de pescado nos permite analizar los posibles problemas que pueden surgir. **Ver figura 6.**

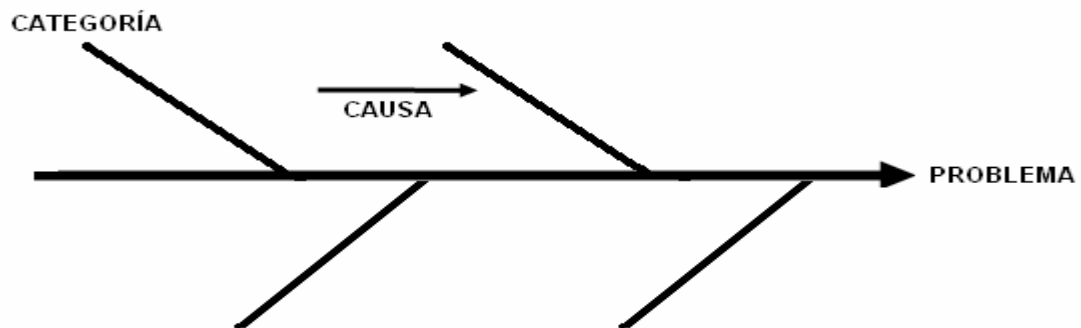


Figura 6: Diagrama espina de pescado

Esta técnica la podemos utilizar conjuntamente con una tormenta de ideas, para ayudarnos a ordenar las posibles soluciones a un problema. Es decir, por un lado generamos ideas y luego utilizamos esta herramienta para organizarlas.

DCO

El objetivo del DCO (Documento de Concepto de Operaciones) es el de comprender el entorno en el cual se encuentra el negocio, describiendo su funcionamiento interno y su relación con el ambiente.

Para la construcción de este documento podemos consultar la plantilla base del DCO, brindada por Software Factory.



Los puntos tratados en el documento básicamente son: análisis del entorno, descripción general, organigrama de la empresa, misión/visión, políticas de desarrollo, de servicios, comercial, de personal, de dirección, etc.

Diagrama de Actividad.

Para representar un proceso de negocio podemos utilizar el diagrama de actividad. El diagrama de actividad o diagrama de proceso, se asemeja a un mapa de procedimientos, mostrando el flujo de actividades: se toman decisiones de acuerdo a las condiciones, para luego pasar a la siguiente actividad o estado. Este modelo también permite representar actividades que ocurren en paralelo, o aquellos casos en los que una única actividad desencadena más de una tarea, o cuando se unen dos o más actividades para formar una tercera.

Casa de calidad o QFD

El esquema QFD (Quality Function Deployment) es una matriz que representa las casas de calidad, en las cuales las filas representan los "qué", o sea, la lista de los requisitos, mientras que las columnas representan los "cómo", es decir, cómo se llevan a cabo los requisitos (casos de uso).

Dado un requisito, se marcan todos los casos de uso que lo implementan y, dado un caso de uso, se marcan todos los requisitos, en los que éste participa.

Debemos recordar que todo requisito, debe ser implementado a través de algún caso de uso y, que todo caso de uso debe satisfacer algún requisito.

Existen numerosas técnicas que se pueden utilizar en cada una de las actividades del proceso de Ingeniería de Requisitos. Y, como toda técnica, es necesario practicar numerosas veces su manejo para lograr sacar el mejor provecho de ellas.

Así mismo, no basta con que utilicemos una sola técnica para lograr nuestros fines: cada una de ellas aporta diferente y vital información sobre el problema en cuestión.



1.8 CONCLUSIONES PARCIALES

Como resultado del estudio realizado sobre la Ingeniería de Requisitos y sus prácticas para el tratamiento de los requisitos, ya que son la base del desarrollo de un producto software, se identificaron las principales actividades que contempla esta ingeniería, extracción, análisis, especificación y validación.

Se analizó el tratamiento de los requisitos en las metodologías de desarrollo que más se utilizan a nivel mundial, RUP y XP. A partir de toda esta bibliografía consultada, se determina que en la propuesta a desarrollar se definirán técnicas para apoyar cada una de las actividades que integran el desarrollo de requisitos.



CAPÍTULO 2: PROPUESTA

2.1 INTRODUCCIÓN

En este capítulo se realiza una descripción de la situación que actualmente posee la F3 en la producción de software. Se muestra una caracterización de los diferentes polos productivos, líneas de desarrollo e investigación de cada uno y los distintos proyectos que los conforman, así como del equipo de desarrollo en cuanto al desarrollo de requisitos. Se argumenta las técnicas seleccionadas que conforma la guía para la aplicación de técnicas para el desarrollo de requisitos. Y finalmente la validación de la misma por criterios de especialistas.

2.2 SITUACIÓN ACTUAL DE LA ORGANIZACIÓN DE LA PRODUCCION DE SOFTWARE EN LA FACULTAD # 3 EN LA UCI.

El proceso de desarrollo de software consiste en un conjunto de acciones y actividades en tiempo y espacio que conforman un sistema, que permite la estructura y desarrollo del software.

La Universidad presta una atención especial a la producción, siendo uno de los objetivos principales el desarrollo de software con calidad, que satisfaga las necesidades de los clientes y pueda ser competitivo en el mercado internacional.

En la F3 de la UCI, actualmente la producción de software se encuentra conformada por tres polos fundamentales y cada uno con una misión y visión e integrado por varios proyectos productivos. Esta información fue extraída del sitio Web de producción-investigación de la F3. La misma no goza de la estabilidad requerida, cambia sistemáticamente.

A continuación realizaremos una descripción de cada polo productivo:



- Polo Gestión de Recursos

Tiene como misión el desarrollo sostenido de proyectos de informatización para las empresas, particularmente sistemas de gestión empresarial y de manejo de información. Especialización en el desarrollo de sistemas para el control de recursos humanos, transporte y agencias de viajes, gestión de información estadística, inventarios, comercio electrónico. Usando las plataformas J2EE, .NET y basado en un modelo estable y de factoría de software.

La visión de este polo se concentra en el desarrollo de aplicaciones empresariales de alta calidad. Se convierte en centro de referencia de la producción de este tipo de software y de consultorías especializadas. Crea y domina frameworks para el desarrollo rápido de aplicaciones con calidad que potencien la reutilización de código. Tales como plataforma de Software AG y GeneXus.

Este polo cuenta con tres líneas de desarrollo y una de investigación, que mencionaremos a continuación: como líneas de desarrollo tenemos, desarrollo de software para la gestión de información estadística, software para la gestión empresarial de recursos y desarrollo de software de apoyo a la comercialización, y como línea de investigación, la investigación en bases de datos avanzadas, réplica de datos, migración entre sistemas gestores de datos.

Se encuentra especializado en un perfil de desarrollo de software de gestión empresarial y en informática para el turismo.

A este polo están vinculados los siguientes proyectos productivos:

- Sistema para la Gestión de recursos humanos.

- Sistema para la Gestión de inventarios.

- Sistema para la Gestión Logística.

- Informatización de la Oficina Nacional de Estadísticas.

- Sistema para la Gestión de Transporte.

- Plataforma para el comercio electrónico.



- Polo productivo Sistemas Legales

Tiene como misión un desarrollo sostenido de proyectos de informatización para el sector jurídico, especialización en el desarrollo de software para la gestión de la actividad registral y la consultoría jurídica. Tomando como bases tecnológicas el desarrollo sobre las plataformas J2EE, .NET. Su visión se concentra en el desarrollo de aplicaciones de gestión jurídica de alta calidad y a la medida. Se convierte en centro de referencia de la producción de este tipo de software y de consultorías especializadas. Crea y domina frameworks para el desarrollo rápido de aplicaciones con calidad que potencien la reutilización de código. Tales como la plataforma de Software AG y GeneXus. Se especializa en la creación de productos clave para la gestión de registros y procedimientos legales. Mantener los registros y automatizar los procedimientos. El desarrollo de proyectos de gestión jurídica con plataformas libres.

Dentro de este las líneas de desarrollo de enfocan al desarrollo de software para la gestión de la actividad registral y el desarrollo de software para la gestión de la consultoría jurídica y como línea de investigación se basa en bases de datos avanzadas y réplica de datos, especializándose en los perfiles de desarrollo de Software de Gestión Empresarial y de informática jurídica y desarrollo de sistemas legales.

Los proyectos incluidos dentro de este polo son, informatización de registros mercantiles inmobiliarios, notarias y registros principales, consultoría Jurídica Internacional, Informatización de la gestión de la dirección de producción asesoría legal IP-UCI.

- Gestión de Proyectos,

Tiene la misión de desarrollar software para la gestión de proyectos, la informatización de la infraestructura productiva y del proceso de producción de software. El desarrollo sostenido de herramientas para la ayuda a la toma de decisiones y con la visión de desarrollar software para la gestión de proyectos.



Polo que dirige la maestría en Gestión de proyectos Informáticos. Domina la tecnología Zope como base tecnológica para la creación de sistemas de gestión de proyectos. Esto trae aparejado el desarrollo de plataformas de productividad para esta tecnología además el desarrollo de componentes y algoritmos para la ayuda a la toma de decisiones. Estos componentes deben tener interfaces para Java, Python y .NET (Mono), por lo que se escribirán en C/C++. Posee una línea de desarrollo de software para la gestión, control y seguimiento de proyecto y tres de investigación, investigación de bases de datos OO (Zope), desarrollo de productos sobre tecnología (Zope-Plone/Phyton) y desarrollo de herramientas para la ayuda a la toma de decisiones e inteligencia artificial.

Como perfiles para este polo se especializa principalmente en el desarrollo de Software de Gestión Empresarial, perfil de Inteligencia Artificial y Reconocimiento de patrones enfoque lógico combinatorio.

Los proyectos a realizar son Informatización del Convenio Cuba-Venezuela, Informatización de la Gestión de la Infraestructura Productiva y una plataforma para la gestión integral de un proyecto productivo.

La F3, poseen entre sus objetivos la integración Formación- Producción- Investigación, contando con una estructura organizativa, que incluye todo un equipo de trabajo para darle cumplimiento a estos objetivos. **En la Fig. 7** se muestra el organigrama de la estructura para conformar el equipo de desarrollo.

En dependencia de la disponibilidad, preparación del personal y de las dimensiones del proyecto a desarrollar, una misma persona puede jugar varios de los roles que se proponen y en un proyecto pueden aplicarse en menor o mayor medida los mismos.

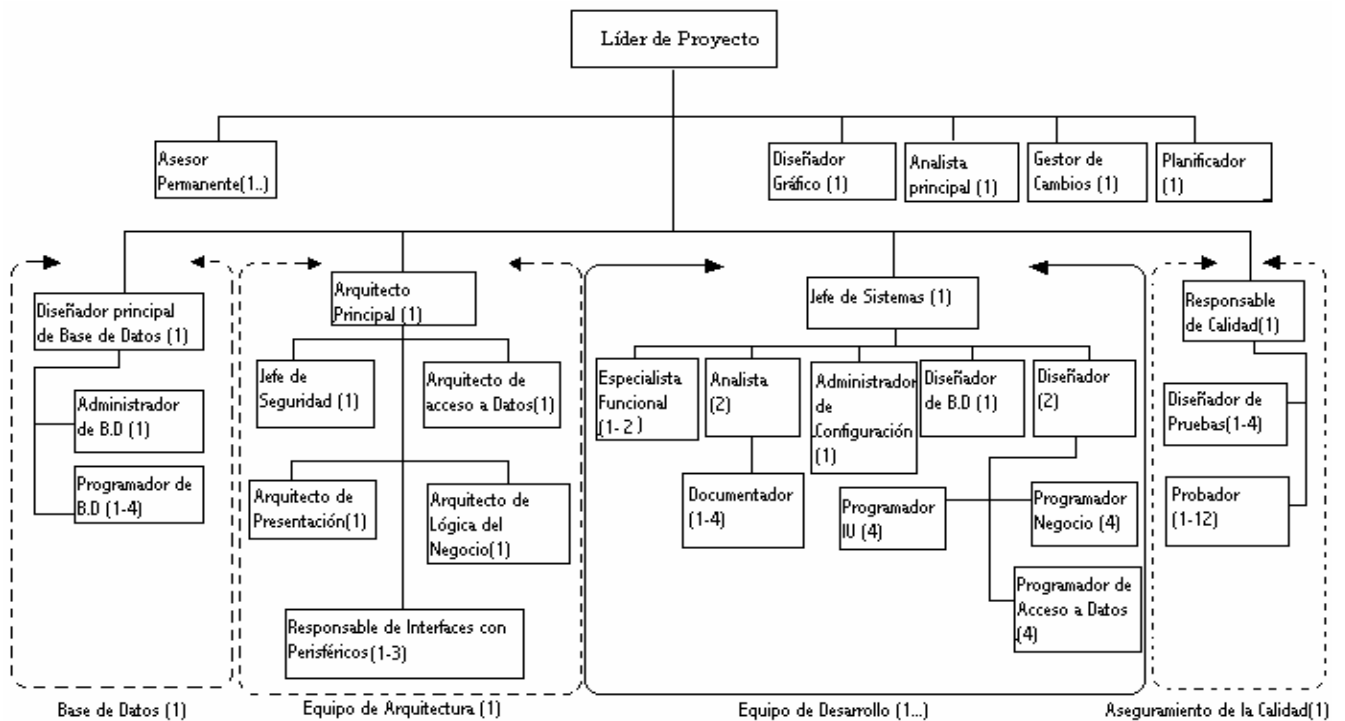


Figura 7: Organigrama propuesto para conformar los equipos de desarrollo de software.

Con el objetivo de comprobar la situación actual del proceso de desarrollo de software en los proyectos de la F3, en cuanto al desarrollo de requisitos en el rol de analista, se realizó una encuesta a través de la confección de un cuestionario (**Anexo 1**) aplicada a un grupo de personas que ejercen el rol de analista en los distintos proyectos de la facultad, principalmente a estudiantes, que son la mayoría los que se encuentran vinculados a la producción. En el cuestionario aplicado se evaluaron varios aspectos y se analizaron principalmente 4 de ellos:

- Preparación para ejercer rol de analista.
- Cursos optativos impartidos por la facultad 3 fueron útiles para el desempeño del rol.
- Uso correcto de las técnicas para el desarrollo de requisitos.
- Conocimiento sobre lo que plantea la Ingeniería de Requisitos.



Para la realización de este diagnóstico se tomó como población los 13 proyectos que desarrolla la facultad # 3 de la UCI, con una muestra del 50% aplicando un total de 20 encuestas a integrantes de dichos proyectos que actualmente ejercen el rol de analista.

De cada polo productivo se seleccionó una muestra de proyectos que actualmente se encuentran en desarrollo y de estos el personal para realizarles la encuesta. Los resultados se reflejan en la **Tabla 1**:

Polo productivo	Nombre del proyecto	Categoría	Cant. Analista Encuestado
Gestión de Recursos	Sistema para la Gestión de inventarios y Almacenes.	Nacional	2
	Informatización de la Oficina Nacional de Estadísticas (ONE)	Nacional	7
	Informatización de la Residencia	(UCI)	5
Sistemas Legales	Consultoría Jurídica Internacional (RN)	Internacional	3
	Tribunales Militares (SITM)	Nacional	2
Gestión de Proyectos	Delfos	Nacional	1

Tabla 2: Muestra seleccionada para la aplicación de la encuesta

Los proyectos de la F3 de la Universidad de las Ciencias Informáticas, se caracterizan en su mayoría por ser proyectos de gestión, dentro de los mismos se encuentran clasificados por el tipo de cliente, es decir se tienen proyectos que poseen clientes nacionales, internacionales e internos, esto último se refieren a proyectos UCI con el objetivo de informatizar la universidad.

A continuación se muestra la **Tabla 2** con los resultados estadísticos porcentuales obtenidos de la aplicación de las encuestas.



Muestra	20 personas para un 100 %
Pregunta 1	
Reciben preparación para ejercer rol de analista.	15%
No reciben preparación para ejercer rol de analista.	85%
Pregunta 2	
Cursos optativos recibidos fueron útiles para el desempeño del rol de analista.	50%
Cursos optativos recibidos no fueron útiles para el desempeño del rol de analista.	50%
Pregunta 3	
Uso correcto de las técnicas para desarrollo de requisitos.	25%
Uso incorrecto de las técnicas para desarrollo de requisitos.	75%
Pregunta 4	
Conocen sobre lo que plantea la Ingeniería de Requisitos.	85%
No conocen sobre lo que plantea la Ingeniería de Requisitos.	15%

Tabla 3: Resultados estadísticos de la encuesta aplicada.

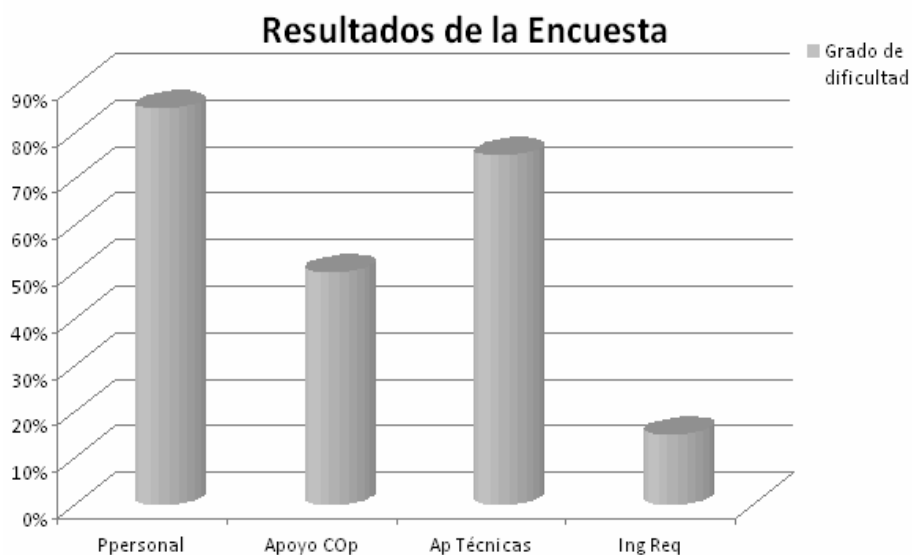


Gráfico 1: Deficiencias en el desarrollo de requisitos.



Leyenda:

Ppersonal: preparación para ejercer rol de analista.

Apoyo COp: Cursos optativos recibidos fueron útiles para el desempeño del rol de analista.

ApTécnicas: Uso correcto de las técnicas de desarrollo de requisitos.

Ing Req: Conocimiento sobre lo que plantea la Ingeniería de Requisitos

Resultados conceptuales

En la tabla 3 se muestran resultados estadísticos obtenidos en la encuesta aplicada y en el **gráfico 1** se representa el porcentaje de deficiencia de los aspectos evaluados.

Por parte de la facultad se han impartido cursos optativos referentes al tema que son útiles y sirven de apoyo para desempeñar con más calidad el rol de analista sin embargo la muestra tomada para la realización de la encuesta dio como resultado que actualmente el personal que ejerce el rol de analista en la F3 no cuenta con una preparación para desarrollar este rol, de lo que se puede decir que este personal puede provocar riesgos en el desarrollo del proyecto, debido a que en un 50% de los estudiantes, los cursos optativos que recibieron no fueron referentes al tema o no le prestaron el interés suficiente.

Para el desarrollo de los requisitos utilizan técnicas de obtención de información, pero no de la forma correcta y solamente las que emplean es con el objetivo de la obtención de requisitos, pudiéndose utilizar en otras actividades del desarrollo de requisitos y además el desconocimiento de aplicación de técnicas para el análisis/especificación y validación de los mismos.

2.3 CARACTERIZACIÓN DEL EQUIPO DE DESARROLLO EN EL DESARROLLO DE REQUISITOS.

Un método utilizado para la investigación fue la entrevista y para su aplicación se realizó una pequeña guía de preguntas. **(Ver Anexo 2)**



De las entrevistas realizadas la mayoría de las personas hizo referencia que para hacer un desarrollo de requisitos eficiente, de las características que deben tener las personas que participen en el proceso, independientemente del rol que ejercen, referente a esto los entrevistados plantearon las siguientes:

- Facilidad de comunicación y negociación.
- Conocer el negocio que se va a informatizar.
- Tener experiencia en la obtención de requisitos.
- Conocer de Ingeniería de Software y de Requisitos.
- Tener la capacidad de identificar los procesos de una organización.
- Conocer la metodología que se vaya a utilizar, sobre tecnologías de desarrollo y los modelos que determinen la calidad del software.
- Tener capacidad para determinar que áreas de la organización están en un cambio transformacional.

Un aspecto fundamental que se obtuvo como fruto de las entrevistas es que debe existir un rol por parte del cliente que sea experto o especialista funcional en el tema objeto de automatización, que aclare todas las dudas que surjan del negocio a automatizar durante la obtención de requisitos y el resto del desarrollo y que participe en las pruebas de calidad que se realicen, este rol como bien dice es llamado especialista funcional.

Después de analizar las características que debe poseer un equipo de desarrollo de software, actualmente en los proyectos productivos de la F3 de la UCI los equipos de desarrollo están caracterizados por:

Al comienzo de cada proyecto los estudiantes que se encuentran ejerciendo el rol de analista no sabían como enfrentarse al cliente correspondiente a su proyecto y como trabajar para el desarrollo y obtención del producto final. Sin embargo en el transcurso de estos últimos 5 años, se ha obtenido experiencias que han servido para la mejora de dichos productos. Además que no poseen una profunda formación en temas de definición de



procesos y en cuanto a las técnicas de recopilación de información que deben utilizar, el conocimiento que se posee no es el suficiente.

2.4 SELECCIÓN DE LAS ACTIVIDADES DEL DESARROLLO DE REQUISITOS (Extracción, Análisis, Especificación, Validación)

Para el desarrollo de la propuesta se tomó lo planteado por Dávila, por definir de una forma clara y precisa el objetivo de cada actividad.

Se toma la actividad de *extracción*, primeramente por poseer un término común y entendible para el estudio de la misma. Representa el comienzo del desarrollo de los requisitos. Aquí, los analistas 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.

El *análisis* porque se considera una actividad sumamente importante y más aún cuando existe un desconocimiento del dominio del negocio. En esta fase se documentan los requisitos acordados con el cliente, en un nivel apropiado de detalle. Se hace un análisis luego de haber elaborado un documento de requisitos inicialmente; aquí se leen los requisitos, se conceptúan, se investigan, se intercambian ideas con el resto del equipo, se resaltan los problemas, se buscan alternativas y soluciones.

La *especificación*: En la práctica, esta etapa se va realizando conjuntamente con el análisis, pero se podría decir que la Especificación es el pasar en limpio el análisis realizado previamente. Es documentar los requisitos acordados con el cliente.

Y la *validación*: la validación de especificaciones realmente significa asegurarse de que el documento de requisitos represente una descripción clara del sistema, y es una verificación final de que los requisitos cubren las necesidades de los usuarios.



Como una conclusión de las actividades por las que está compuesto el desarrollo de los requisitos se puede plantear que el análisis y la especificación funcionan como una sola por que en la práctica las dos se desarrollan simultáneamente, en la propuesta se definen técnicas comunes que se aplicarán en dicha actividad, análisis/especificación.

2.5 PROPUESTA: GUÍA PARA LA APLICACIÓN DE TÉCNICAS PARA EL DESARROLLO DE REQUISITOS.

Para realizar este proceso, no existe una única técnica estandarizada y estructurada que ofrezca un marco de desarrollo que garantice la calidad del resultado. Existe en cambio un conjunto de técnicas, se debe tener en cuenta la selección de las mismas y el éxito de los resultados que se obtengan, depende en gran medida tanto del equipo de análisis y desarrollo, como de los propios clientes o usuarios que en ella participen.

Dentro del proceso de desarrollo de software una fase primordial para garantizar el éxito de los proyectos es la Ingeniería de Requisitos y un componente fundamental dentro de ésta es el desarrollo de los requisitos. La propuesta de una guía para la aplicación de técnicas para el desarrollo de requisitos en los proyectos de la F3 está basada en todo el estudio que se realizó en el capítulo anterior definiendo como base para la propuesta.

Existen un conjunto de técnicas para el desarrollo de requisitos, que se utilizan en cada una de sus actividades, es de gran importancia la aplicación de las mismas para así lograr la mayor cantidad de información y con la calidad requerida.

La metodología a utilizar queda a la elección del líder del proyecto que es el encargado de seleccionar la metodología de desarrollo, dependiendo del tipo de complejidad del proyecto a desarrollar.

Las actividades de extracción, análisis/especificación, validación definas por la Ingeniería de Requisitos se reflejan en la propuesta de procedimiento.



2.5.1 Guía para la aplicación de técnicas para el desarrollo de requisitos.



Figura 8: Pasos para la aplicación de la Guía

Descripción de los pasos para la aplicación de la guía.

A continuación se realizará una descripción de los pasos para la aplicación de técnicas para el desarrollo de requisitos mostrado en la **figura 8**.



Inicialmente debe conformarse un grupo de trabajo del proyecto para la realización de las actividades vinculadas al desarrollo de requisitos, así como dar a conocer el cliente o los clientes de la aplicación a desarrollar.

Los proyectos de la facultad # 3 de la UCI se encuentran clasificados por tipo de cliente, en internacionales, nacionales e internos (UCI) (**Ver figura 9**), estos últimos son proyectos para la informatización de la universidad. Se realiza esta clasificación porque en el momento de aplicar las técnicas para el desarrollo de requisitos, estas no se ejecutan de la misma forma.

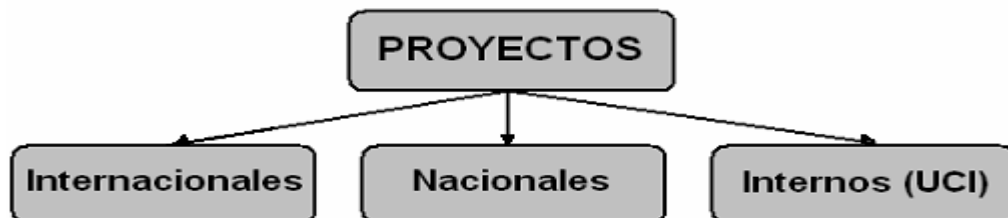


Figura 9: Clasificación de proyectos según el tipo de cliente.

Los proyectos internacionales se caracterizan por tener cliente a distancia lo que dificulta una interacción personal con los mismos. Para aplicar las técnicas, en este caso es necesario el uso de las Tecnologías de la Informática y las Comunicaciones (TIC), se puede citar opciones como, foros montados en aplicaciones web donde interactúen todos los futuros clientes de la aplicación con el objetivo de conocer sus necesidades y expectativas, el uso de videos conferencias orientados a este tipo de actividad, el o los clientes pueden redactar posibles requisitos y enviarlos por correo electrónico, entre otras.

Con respecto a los proyectos nacionales existe mayor probabilidad de planificar una cita con los clientes e interactuar personalmente, aunque no se desecha el uso de las TIC en caso que la entidad a automatizar posea condiciones tecnológicas.



Y los proyectos internos de la universidad tienen la característica de que los clientes poseen una alta probabilidad de estar presente por formar parte del personal de trabajadores del centro lo que posibilita una excelente comunicación entre clientes y desarrolladores.

Antes de comenzar a realizar el desarrollo de los requisitos, es de gran importancia aplicar la técnica de la entrevista con el con el objetivo de identificar los interesados en el sistema y las características generales del negocio que se desea informatizar, además de poder descubrir cuál es el alcance del proyecto, quienes son los beneficiados, se trata de que las preguntas conlleven a una respuesta abierta por parte del cliente para tratar de recopilar la mayora cantidad de información.



Figura 10

Previo a las actividades del desarrollo de requisitos, es importante tener una preparación del equipo de desarrollo (**Figura 10**) con la posibilidad de mejorar la comunicación entre clientes y desarrolladores. Esta preparación consiste en un estudio preliminar sobre los términos y procesos que tienen lugar en el ambiente donde se va a aplicar el proyecto, y para esta se aplicarán las técnicas de:

- Observación: Se utiliza con mayor frecuencia en proyectos internos y nacionales. Se aplica durante la actividad práctica. Un elemento importante es que el grupo de trabajo creado debe interactuar de forma directa, ejecutando los procesos que tienen lugar, con el objetivo de comprender el funcionamiento de los mismos desde una perspectiva interna del negocio. El objetivo de su aplicación es el de observar (valga



la redundancia) como se realiza el flujo de trabajo, es una forma de entender de cómo se trabaja en una organización determinada.

- Arqueología de documentos: Es importante la aplicación de esta técnica durante esta etapa ya que se debe analizar a profundidad toda la base documental dada por la organización para poder obtener información y que el trabajo sea más provechoso.

EXTRACCIÓN

La extracción tiene como objetivo general la captura de los requisitos planteados por los clientes. En la **figura11** se representa esta actividad con las técnicas a aplicar.



Figura 11: Técnicas para la actividad de extracción

Técnicas para la actividad de extracción

Tormenta de ideas: La intención en su aplicación es la de generar la máxima cantidad posible de requisitos para el sistema. En esta técnica todas las ideas tomadas en una primera instancia son válidas. En esta actividad pueden participar todas aquellas personas



que se encuentren afectados por la aplicación (clientes), es decir los que la utilizarán y el personal experimentado del equipo de desarrollo.

La entrevista: Se realiza con el objetivo de emplearla para recolectar información en forma verbal, a través de preguntas elaboradas por el analista, estas preguntas se les realizan a quienes se encuentren afectados por la aplicación, a usuarios con gran nivel de conocimiento del sistema o a personas que pueden proporcionar datos. Las entrevistas se pueden realizar en grupo o individual. La entrevista no es una forma de interrogación, más bien es una forma de conversación, por este método los analistas pueden recaudar información que no es posible recopilar de otra manera.

Una vez delimitado las áreas donde se implantara la aplicación se interactúa con el directivo responsable de la misma. Obteniendo una información más viable de los requisitos. En caso que la entrevista no sea personal, entonces se recomienda el uso de las TIC.

Cuestionarios: Se aplica para obtener información de los clientes, se pueden aplicar dos tipos de cuestionarios, los de preguntas abiertas y los de preguntas cerradas, se aplica uno de los dos de acuerdo a las necesidades del equipo de desarrollo. Esta técnica es un complemento de la entrevista, se hace uso de ella para una entrevista más precisa y directa.

Resultado: Se obtiene un documento preliminar de requisitos.

Nota: En ninguna actividad se debe hacer uso de una sola técnica para la obtención de información, porque para lograr un buen desarrollo de los requisitos es de mucha ayuda una combinación de las mismas.

ANÁLISIS / ESPECIFICACIÓN

En esta actividad principalmente se definen los requisitos funcionales y los no funcionales, se integran en un documento de especificación, el cual se puede ir realizando a la vez



donde se realiza una descripción formal del requisito, que finalmente el sistema a construir, va a cumplir. En la **figura 12** se muestra una representación de las técnicas a aplicar e esta actividad.

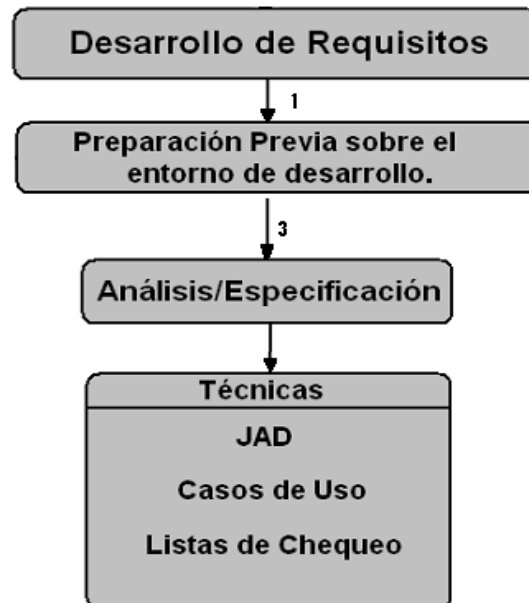


Figura 12: Técnicas para la actividad de análisis/especificación

Técnicas para la actividad de análisis/especificación.

Diseño de Aplicaciones Conjuntas (JAD): Es una técnica que se deriva de la entrevista pero se aplica de forma grupal. La misma se aplica con el objetivo de entender cuál es el problema que se quiere resolver y cuáles son las posibles soluciones que se le dan al mismo, las ideas se expondrían en grupo para que cada integrante pueda dar su opinión y analizar las mismas entre todos, con la utilización de esta técnica son mayores las posibilidades de que el problema y las soluciones a los mismos se ajustan en la mejor medida a las necesidades y requisitos de los usuarios e involucrados.

Con el objetivo de reunir a un grupo definido por parte del equipo de desarrollo y del cliente para analizar los requisitos extraídos en la fase anterior a un nivel apropiado de detalle del documento preliminar.



Lista de chequeos: Esta técnica se utiliza en las actividades de revisión interna que realiza el equipo de desarrollo con el objetivo de realizar una revisión a cada uno de los documentos que deben ser entregados al cliente, donde los mismos deben cumplir con las condiciones que se describen en la lista de chequeos.

Casos de Uso: Se utiliza esta técnica por ser la más usada actualmente en los proyectos, además facilita el análisis / especificación de los requisitos porque está diseñada para representar y especificar el comportamiento y funcionalidades de un sistema.

Resultado: Se genera un glosario de términos con el objetivo de ser utilizado durante casi todo el proceso, que contenga aquellos términos referentes al negocio a informatizar para que el equipo de desarrollo tenga un conocimiento profundo de los mismos y el proceso sea más exitoso y del documento de especificación actualizado para ser validado por el cliente.

Plantilla del documento de especificación de requisitos preliminar. (Tabla 4)

Primera Cuartilla				
Documentos de Especificación de Requisitos Preliminar (DERP).				
Segunda cuartilla				
	Nombre y Apellidos	Cargo	Organización	Firma
Elaborado por:				
Participantes:				
Aprobado por:				



Observaciones
Tercera cuartilla
1. Objetivo Constituir una referencia general a tener en cuenta por los jefes y especialistas para realizar la Especificación Preliminar de Requisitos, producir adecuadamente la definición de los procesos actuales, el Glosario de Términos, los Casos de Uso, logrando así un desarrollo de requisitos eficiente con calidad, un lenguaje común entre el Cliente y Desarrollador, una descripción sencilla y comprensible por parte del Cliente de sus procesos, y el establecimiento y mantenimiento correcto de los acuerdos oficiales con el Cliente.
2. Alcance Se aplica en el Proceso de desarrollo de requisitos de todos los proyectos de desarrollo de software que ejecute la organización, en este caso la F3 de la UCI.
3. Definiciones y Términos Las definiciones de los términos más importantes empleados se encuentran en el documento Glosario.
4. Descripción del proceso La estructura del Proceso de Especificación Preliminar de Requisitos consta de las siguientes actividades: a) Estudio del dominio del problema. b) Elaboración del Glosario de Términos. c) Especificación de Casos de Uso.

Tabla 4: Formato del documento de especificación de requisitos preliminar



VALIDACIÓN

El objetivo de esta actividad es obtener una validación de los requisitos propuestos, eliminando inconsistencia y ambigüedades que se puedan presentar. En la **figura 13** se muestra una representación de las técnicas a aplicar e esta actividad.

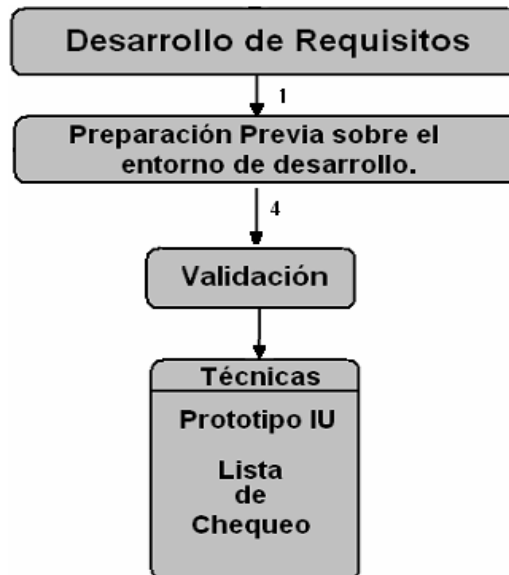


Figura 13: Técnicas para la actividad de validación

Técnicas para la actividad de validación

Prototipo: Se utiliza esta técnica por ser los prototipos simulaciones del posible producto, que luego son utilizados por el usuario final, permitiéndonos conseguir una importante retroalimentación en cuanto a si el sistema diseñado en base a los requisitos recolectados le permite al usuario realizar su trabajo de manera eficiente y efectiva.

Se desarrolla un prototipo de interfaz de usuario (IU) a partir del documento de especificación de requisitos, con el objetivo de dar una visión de cómo debe quedar el software y a través de este se obtiene una validación de los requisitos.

Lista de Chequeo: En esta actividad se realiza una continuidad de revisión de cada uno de los documentos que deben ser entregados al cliente.



Resultado: Se obtiene una validación del documento de especificación de requisitos con el objetivo de dar paso a otra fase del desarrollo del software.

2.6 VALIDACIÓN DE LA PROPUESTA

La propuesta de guía fue validada por el método de criterios a especialistas, del centro Desoft (Desarrollo de software) División Camagüey y del ICID (Instituto Central de Investigaciones Digital. Para la realización de esta validación se conformó una serie de preguntas en las cuales los especialistas dieron su criterio y recomendaciones para su mejora. Las preguntas para realizar la validación fueron las siguientes:

1. ¿Cree que exista un orden lógico en cuando al desarrollo de requisitos planteado?
2. ¿Las actividades planteadas responden al tratamiento por el que deben transitar los requisitos?
3. ¿Cree que son suficientes las técnicas que se plantean en la guía?

Criterio # 1:

Especialista: Eugenia Muñiz Lodos, ocupa el cargo de Investigadora Auxiliar con más de 30 años de experiencia haciendo software, perteneciente al Instituto Central de Investigación Digital (ICID).

Es conocida la repercusión de la Ingeniería de Requisitos en la calidad del producto de software.

La guía que se plantea, incluye las actividades más importantes del desarrollo de requisitos y contempla las técnicas más utilizadas de las mismas. Es un aproximación inicial que debe ser probada en algún proyecto de desarrollo de software que se inicie y de acuerdo a los resultados y su utilización pudiera ser mejorada o adecuada a tipos específicos de aplicaciones. Es una buena idea para orientar este proceso.



Criterio # 2:

Especialista: Lic. José Manuel Saldívar García, ocupa el cargo de analista con 4 años de experiencia haciendo software, perteneciente a Desoft (División Camagüey) categorizado por Especialista B en Ciencias Informáticas.

La guía para la aplicación de técnicas para el desarrollo de requisitos que se plantea describe las características necesarias para este proceso con un orden lógico, refleja las técnicas más usadas en el desarrollo de los requisitos, principales para el trabajo con estos. En su opinión refleja que es un estudio acertado.

Recomienda que se ponga en práctica a nuevos proyectos, es decir a proyectos que comiencen desde cero para valorar los resultados y ver si es necesario una mejora de la misma. Además de que se pueden incluir más técnicas, pero para un comienzo las técnicas propuestas son acertadas.

Criterio # 3:

Especialista: Lic. Sivert Martínez Ramírez, ocupa el cargo de Jefe de Proyecto Offimant con 6 años de experiencia haciendo software, perteneciente a Desoft (División Camagüey) categorizado por Especialista Principal en Informática.

La Ingeniería de Requisitos como antesala del desarrollo de un proyecto, es la etapa fundamental donde se definen las características del sistema, así como el alcance del mismo, y se estructuran las necesidades del cliente, por lo que se hace esencial para lograr la calidad del producto.

La guía que se plantea agrupa una serie de actividades necesarias y con un orden lógico para el desarrollo de este proceso, además de que plasma una serie de técnicas usadas en el mismo que son comunes y eficientes a la vez para el trabajo con requisitos. Opino que es un estudio acertado con grandes perspectivas, ideal para el desarrollo de cualquier software que se inicie, con grande posibilidades de aplicación a la medida, según las aplicaciones que se tengan.



Estudio que debe ser tomado como base ante la ejecución de las aplicaciones en cualquier entidad desarrolladora de software.

Se recomienda después que se ponga en práctica y se valoren los resultados, enriquecer la guía con nuevas técnicas, las cuales sean adaptables a las condiciones del equipo de desarrollo.

2.7 CONCLUSIONES PARCIALES

- Con el estudio realizado a los proyectos productivos de la F3 de la UCI con respecto al desarrollo de los requisitos se detectó de manera preliminar las dificultades inherentes a estos procesos.
- Se hizo una selección de las técnicas para el desarrollo de requisitos que se ajustaban a las condiciones de los proyectos de la facultad con el objetivo de realizar una guía para la aplicación de las mismas para el desarrollo de los requisitos en los proyectos, objetivo que se le dio cumplimiento.
- Finalmente se realizó validaciones de la guía propuesta arrojando resultados satisfactorios.



CONCLUSIONES GENERALES

- La propuesta de utilización de las técnicas de desarrollo de requisitos garantiza una mayor visión de los requisitos necesarios para los clientes y un entendimiento detallado en los miembros del equipo de desarrollo.
- Con el diagnóstico realizado a los proyectos productivos de la F3 de la UCI con respecto al desarrollo de requisitos, se detectó limitaciones en la utilización de las técnicas.
- Se elaboró una guía de técnicas para orientar el trabajo de los analistas, incluyendo todas las actividades del desarrollo de requisitos y técnicas para la aplicación de las mismas.
- La guía fue validada por especialistas dando un criterio satisfactorio.



RECOMENDACIONES

- Poner en práctica la guía para algún proyecto que se inicie y darle continuidad.
- Continuar el desarrollo de esta propuesta haciéndola cada vez más útil para ser usada en el desarrollo de requisitos en proyectos productivos.
- Aplicar la propuesta con el fin de que se valide.
- Hacer de esta investigación un material de consulta del personal que ejerza como analista en los proyectos.



REFERENCIAS BIBLIOGRÁFICAS

- [1] Llanes M A. Fbioinformática/Glosario. Glosario. Universidad de La Habana. Abril 2005. Junio 2007
<<http://fbio.uh.cu/bioinfo/glosario.html>>
- [2] Xpress Hosting. Web Hosting en México. Glosario.Copyright 2007. Junio 2007
<http://www.xpress.com.mx/glosario_s.php>
- [3] Pressman S R. Ingenieria del Software. Un enfoque práctico. 5ta Edición. La Habana 2005. pág 3 Volumen 1.
- [4] Zavala R J. La Ingeniaria del Software. Septiembre 2002. Junio 2007
<<http://www.angelfire.com/scifi/jzavalar/apuntes/IngSoftware.html>>
- [5] Jacobson, I., Booch, G., Rumbaugh J., El Proceso Unificado de Desarrollo de Software, Addison Wesley 2000.
- [6] Jacobson, I. 1998. "Applying UML in The Unified Process" Presentación. Rational Software.
<[http://www.rational.com/uml como UMLconf.zip](http://www.rational.com/uml%20como%20UMLconf.zip)>
- [7] Meilir Page-Jones. The Practical Guide to Structured Systems Design. Yourdon Press, 1980.
- [8] Young R R. The Requirements Engineering Handbook. 2004
<www.artechhouse.com>
- [9] CMMI Glosario
<<http://www.grupoconsultoria.com.co/glosario.htm>>
- [10] Ortas (1997). Aproximación a la Ingeniería de Requerimientos. Uruguay. Universidad ORT Uruguay.
- [11] Wiegers E K. More About Software Requirements: Thorny Issues and Practical Advice. 2006.
- [12] Lavariega, 2005 G. Kotonya, I. Sommerville (Agosto 2005). "Requirements Engineering: Processes and Techniques"
- [13] Escalona, Koch.(Sevilla, dic 2002).Ingeniería de Requisitos en Aplicaciones para la Web –Un estudio comparativo.



-
- [14] Lopez, dic 1999. Análisis de Requerimientos de Software. Estado del Arte.
- [15] Davyt D, N. (2001). Ingeniería de requerimientos. Una guía para extraer, analizar, especificar y validar los requerimientos de un proyecto.
- [16] José H. Canós, Patricio Letelier y M^a Carmen Penadés. Metodologías Ágiles en el Desarrollo de Software.
- [17] SCRUM t's About Common Sense. 2007
<<http://www.controlchaos.com>>
- [18] DSDM
<<http://www.dsdm.org>>
- [19] Agile Project Management. Adaptive Software Development. 2006
<<http://www.adaptivesd.com>>
- [20] Feature Driven Development.
<<http://www.featuredrivendevelopment.com/>>
- [21] Craig, L. (1999). UML Y PATRONES. Introducción al análisis y diseño orientado a objetos. España, Pearson.
- [22] [Brooks87] Frederik P. Brooks - No Silver Bullet. Essence and Accidents in Software Engineering. IEEE Computer. Abril 1987.



BIBLIOGRAFÍA

1. Booch, G. y Jacobson, I. RumbaughT, J. El Lenguaje Unificado de Modelado. España: Addison Wesley, 1999.
2. Brooks P F. - No Silver Bullet. Essence and Accidents in Software Engineering. IEEE Computer. Abril 1987.
3. Dávila, Nicolás Davyt. INGENIERÍA DE REQUERIMIENTOS UNA GUÍA PARA EXTRAER, ANALIZAR, ESPECIFICAR Y VALIDAR LOS REQUERIMIENTOS DE UN PROYECTO. Uruguay: Facultad de Ingeniería UNIVERSIDAD ORT URUGUAY.
4. Escalona M.J., Koch N. Ingeniería de Requisitos en Aplicaciones para la Web: Un estudio comparativo. España: Universidad de Sevilla. Lenguajes y Sistemas Informáticos.
5. Escribano, Gerardo Fernández. Introducción a Extreme Programming, Ingeniería del Software II.
6. IEEE Recommended Practice for Software Requirements Specifications, 25 June 1998 IEEE-SA Standards Board
7. José H. Canós, Patricio Letelier y Maria Carmen Penadés. Metodologías Ágiles en el Desarrollo de Software. s.l. : DSIC -Universidad Politécnica de Valencia.
8. Larman, Craig. UML y Patrones. Introducción al Análisis y Diseño Orientado a Objeto. S.I.: Prentice Hall.
9. López, dic 1999. Análisis de Requerimientos de Software. Estado del Arte.
10. Mcdonald (Septiembre del 2005). Definición de Perfiles en Herramientas de Gestión de Requisitos
11. Ortas (1997). Aproximación a la Ingeniería de Requerimientos. Uruguay. Universidad ORT Uruguay.
12. Pressman, R. INGENIERÍA DEL SOFTWARE: Un enfoque práctico. España: McGraw-Hill.
13. Sommerville, G. Kotonya e I. Ingeniería de Requerimientos. Introducción a Ingeniería de Requisitos.



14. Wesley Addison. Mastering the Requirements Process. Second Edition. March 17, 2006.
15. Wiegers E Karl. Software Requirements, Second Edition. 2003



GLOSARIO DE TÉRMINOS

A

Analista: El analista es un miembro del equipo de desarrollo técnico (ingenieros en sistemas o analistas de negocios, que desarrollan los requerimientos del sistema) que está capacitado para definir problemas y analizar, desarrollar y expresar algoritmos.

Aprendiz: Persona que aprende un oficio.

C

Cliente: La entidad o entidades para los cuales los requerimientos deben ser satisfechos en el sistema que está siendo definido y desarrollado. Éste puede ser el usuario final de un sistema terminado. Es la persona (s) que pagan por el producto y normalmente (pero no necesariamente) define los requisitos. En la práctica el cliente y el proveedor pueden ser miembros de la misma organización. Ésta es la entidad para la cual el desarrollador de sistema debe demostrar que el sistema terminado satisface los requerimientos especificados.

Cuestionario: Lista de preguntas que se proponen con cualquier fin.

D

Documentos ESRE: Documento de Especificaciones de Requisitos.

F

Framework: En el desarrollo de software, un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, librerías y un lenguaje de scripting entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.



G

Guía: Se define como el documento que describe en forma sistemática los objetivos de las diferentes técnicas para su aplicación.

I

IEEE: Institute of Electrical and Electronics Engineers: Importante asociación de técnicos y profesionales, con sede en los Estados Unidos. Fue fundada en 1884 y favorece la investigación en campos diversos, como la tecnología aeroespacial, la computación, las comunicaciones y la tecnología biomédica. Promueve la estandarización de normas.

IP: Infraestructura Productiva

L

Línea Base: Una especificación o sistema que ha sido formalmente revisado y aprobado, que sirve como base para un desarrollo posterior y que puede ser modificado sólo a través de procedimientos formales de control de cambios (EEEE Std 610.12-1990).

O

Organización: La organización es una unidad social coordinada, consciente, compuesta por dos personas o más, que funciona con relativa constancia a efecto de alcanzar una meta o una serie de metas comunes. Según esta definición, las empresas productoras, los organismos de los gobiernos locales, estatales y federal son organizaciones.

P

Prototipo: Es un modelo a escala o imitación de lo real, pero no tan funcional para que equivalga a un ya producto final, que no lleva a cabo la totalidad de las funciones necesarias del sistema final. Proporcionando una retroalimentación temprana por parte de los usuarios acerca del Sistema.



Plataforma: una plataforma es precisamente el basamento, ya sea de hardware o software, sobre el cual un programa puede ejecutarse. Ejemplos típicos incluyen: arquitectura de hardware, sistema operativo, lenguajes de programación y sus librerías de tiempo de ejecución.

R

Requisito: Una condición o capacidad necesaria para un usuario para resolver un problema o alcanzar un objetivo.

Requisito funcional: Alguna actividad o servicio que debe realizar el software y que puede ser comprobada.

Requisito no funcional: Limitación de hardware o software bajo la cual el sistema debe operar.

Rol: Un conjunto de expectativas de conducta asociadas a una persona, un patrón de comportamiento que se espera de quién desempeñe cada puesto, con cierta independencia de la persona que sea.

S

STD: Estándares

T

Técnicas: Sucesión ordenada de acciones que se dirigen a un fin concreto, conocido y que conduce a unos resultados precisos.

U

UCI: Universidad de las Ciencias Informáticas

Usuario: Aquella persona o grupo de personas dentro de la empresa que utilizarán el software desarrollado, incluido el tomador de decisiones.



Usuario Final: Toda aquella persona u organización siendo influenciada o ejerciendo influencia sobre el software que está siendo construido.

