

PROCEDIMIENTO PARA MANTENER EL SEGUIMIENTO DE LOS  
REQUISITOS DE SOFTWARE EN EL CENTRO DE INFORMATIZACIÓN PARA  
LA GESTIÓN DE ENTIDADES

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS  
INFORMÁTICAS

**AUTOR**

Adrienne García Rodríguez

**TUTORES**

Ing. Sandy Machado Scull

Ing. Yanet Vega Miniet

Ciudad de La Habana, 2010

“Año 52 de la Revolución”

## DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo al Centro de Informatización para la Gestión de Entidades de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

\_\_\_\_\_  
Adrienne García Rodríguez

\_\_\_\_\_  
Ing. Sandy Machado Scull

\_\_\_\_\_  
Ing. Yanet Vega Miniet

## RESUMEN

En la actualidad, el Centro de Informatización para la Gestión de Entidades no cuenta con un procedimiento definido que posibilite realizar de forma óptima el seguimiento a los requisitos, lo que influye directamente en el eficiente desarrollo de un proyecto de software.

En el presente trabajo de diploma se define un procedimiento para mantener el seguimiento a los requisitos en dicho centro. Para ello se realiza un estudio de la Ingeniería de requisitos, específicamente del Seguimiento a los requisitos, con el fin de identificar las tendencias que en la actualidad existen relacionadas con el tema y así elaborar una propuesta válida para ser implementada en el centro.

El diseño del procedimiento cuenta con ocho pasos, en cada uno se definen roles, artefactos y actividades.

El resultado fundamental se centra en la obtención de un procedimiento cuya aplicación garantice realizar eficientemente el seguimiento a los requisitos.

Para validar el procedimiento se aplicó en el proyecto Dirección por Objetivos de la línea Planificación del Centro de Informatización para la Gestión de Entidades.

### PALABRAS CLAVE

Seguimiento a los requisitos, elementos de seguimiento, trazabilidad.

# ÍNDICE DE CONTENIDOS

DECLARACIÓN DE AUTORÍA.....	I
RESUMEN.....	II
INTRODUCCIÓN.....	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA .....	5
1.1. INTRODUCCIÓN.....	5
1.2. PROCESO DE DESARROLLO DE SOFTWARE .....	5
1.3. REQUISITO DE SOFTWARE.....	5
1.3.1. REQUISITOS FUNCIONALES Y REQUISITOS NO FUNCIONALES .....	7
1.4. INGENIERÍA DE REQUISITOS.....	7
1.4.1. ACTIVIDADES EN LA INGENIERÍA DE REQUISITOS .....	8
1.4.2. CARACTERÍSTICAS DE LA INGENIERÍA DE REQUISITOS .....	9
1.4.3. IMPORTANCIA DE LA INGENIERÍA DE REQUISITOS.....	10
1.5. GESTIÓN DE REQUISITOS.....	10
1.6. SEGUIMIENTO DE LOS REQUISITOS .....	11
1.6.1. MATRICES DE SEGUIMIENTO.....	12
1.7. HERRAMIENTAS.....	12
1.7.1. GATHERSPACE.....	13
1.7.2. SOFTWARE PLANNER 9.0.....	13
1.7.3. REM (REQUIREMENTS MANAGEMENT).....	13

1.7.4.	OSRMT (OPEN SOURCE REQUIREMENTS MANAGEMENT TOOL).....	13
1.7.5.	REQUISITEPRO.....	14
1.7.6.	ENTERPRISE ARCHITECT (EA).....	14
1.8.	CONCLUSIONES PARCIALES.....	15
CAPÍTULO 2. PROPUESTA DE SOLUCIÓN.....		16
2.1	INTRODUCCIÓN.....	16
2.2	CARACTERÍSTICAS DEL PROCEDIMIENTO .....	16
2.3	ROLES .....	16
2.4	ROLES POR ACTIVIDADES .....	17
2.5	PROCEDIMIENTO PARA EL SEGUIMIENTO A LOS REQUISITOS. ESTRUCTURA.....	19
2.5.1	PASO 1. IDENTIFICAR ELEMENTOS DE SEGUIMIENTO.....	19
2.5.2	PASO 2. DEFINIR LAS RELACIONES ENTRE LOS ELEMENTOS DE SEGUIMIENTO .....	21
2.5.3	PASO 3. ESPECIFICAR LOS ATRIBUTOS DE CADA ELEMENTO DE SEGUIMIENTO.....	24
2.5.4	PASO 4. INSTALAR Y CONFIGURAR LA HERRAMIENTA DE GESTIÓN DE REQUISITOS.....	41
2.5.5	PASO 5. REGISTRAR ELEMENTOS DE SEGUIMIENTO A LOS REQUISITOS.....	41
2.5.6	PASO 6. DEFINIR LAS DEPENDENCIAS ENTRE LOS ELEMENTOS DE SEGUIMIENTO .....	42
2.5.7	PASO 7. CREAR LA LÍNEA BASE DE LOS ELEMENTOS DE SEGUIMIENTO .....	43
2.5.8	PASO 8. OBTENER INFORMACIÓN DE SEGUIMIENTO.....	49
2.6	CONCLUSIONES PARCIALES.....	50
CAPÍTULO 3. VALIDACIÓN DEL PROCEDIMIENTO.....		51

3.1	INTRODUCCIÓN.....	51
3.2	APLICACIÓN DEL PROCEDIMIENTO.....	51
3.3	VALIDACIÓN DEL PROCEDIMIENTO.....	56
3.4	CONCLUSIONES PARCIALES.....	59
	CONCLUSIONES .....	60
	RECOMENDACIONES .....	61
	BIBLIOGRAFÍA.....	62
	ANEXO 1. FORMULARIO DE SOLICITUD DE CAMBIO.....	64
	ANEXO 2. REPORTE APROBACIÓN: .....	67
	ANEXO 3. ENCUESTA A LOS PARTICIPANTES EN LA APLICACIÓN DEL PROCEDIMIENTO. ....	69
	ANEXO 4. PLANTILLA DE REQUISITOS RECHAZADOS.....	70
	ANEXO 5. REGISTRO DE REVISIONES Y CAMBIOS.....	72
	ANEXO 6. PLANTILLA DE REGISTRO DE INCONSISTENCIAS.....	73

## ÍNDICE DE FIGURAS

FIGURA 1. ROLES POR ACTIVIDADES. ....	18
FIGURA 2. RELACIONES ENTRE LOS ELEMENTOS DE SEGUIMIENTO.....	22
FIGURA 3. PROCESO DE CONTROL DE CAMBIO SEGÚN PRESSMAN. ....	44
FIGURA 4. FLUJO DE ACTIVIDADES CREAR LÍNEA BASE. ....	46
FIGURA 5. FLUJO DE ACTIVIDADES CONTROL DE CAMBIO.....	48
FIGURA 6. UTILIDAD DE COMPARACIÓN. RESULTADOS.....	54

FIGURA 7. MATRIZ DE RELACIONES PROCESOS DE NEGOCIO Y GESTIONAR OBJETIVOS .....55

FIGURA 8. MATRIZ DE RELACIONES COMPONENTES Y GESTIONAR GRUPOS DE USUARIOS.....56

## ÍNDICE DE TABLAS

TABLA 1. ROLES .....16

TABLA 2. PASO 1. IDENTIFICAR ELEMENTOS DE SEGUIMIENTO.....19

TABLA 3. ELEMENTOS DE SEGUIMIENTO. ....20

TABLA 4. PASO 2. DEFINIR LAS RELACIONES ENTRE LOS ELEMENTOS DE SEGUIMIENTO.....21

TABLA 5. PASO 3. ESPECIFICAR ATRIBUTOS DE CADA ELEMENTO DE SEGUIMIENTO. ....24

TABLA 6. NECESIDADES DE LA ORGANIZACIÓN. ....24

TABLA 7. PROCESO DE NEGOCIO. ....27

TABLA 8. AGRUPACIÓN DE REQUISITOS. ....29

TABLA 9. REQUISITOS FUNCIONALES .....30

TABLA 10. REQUISITOS NO FUNCIONALES.....33

TABLA 11. COMPONENTE. ....36

TABLA 12. SOLICITUD DE CAMBIO.....39

TABLA 13. PASO 4. REGISTRAR ELEMENTOS DE SEGUIMIENTO. ....41

TABLA 14. PASO 5. DEFINIR LAS DEPENDENCIAS ENTRE LOS ELEMENTOS DE SEGUIMIENTO.....43

TABLA 15. PASO 7. OBTENER INFORMACIÓN DE SEGUIMIENTO.....49

TABLA 16. ELEMENTOS DE SEGUIMIENTO SELECCIONADOS.....52

## INTRODUCCIÓN

Actualmente la producción de software fiable, de calidad y de fácil mantenimiento, es todavía uno de los retos pendientes de la Ingeniería de Software. La mayoría de los errores cometidos provienen de las primeras fases de su ciclo de vida, debido a que el costo de resolver estos problemas crece exponencialmente según avanza el proyecto, resulta de vital importancia centrarse en la mejora de las primeras etapas, en lugar de postergarlo a etapas más tardías. La mayor parte de los enfoques sobre estos temas debería centrarse en los requisitos, los cuales son de suma importancia dentro del proceso de desarrollo del software.

Hoy en día se orientan los esfuerzos hacia el entrenamiento, diagnóstico, consultoría y formación según la demanda de los proyectos productivos de la Universidad de las Ciencias Informáticas (UCI), los mismos se ven obligados a realizar grandes cambios en productos prácticamente terminados, debido a inconformidades con las solicitudes de los clientes que en etapas iniciales no fueron entendidas correctamente. Los clientes, por su parte, no siempre presentan la preparación y el conocimiento requeridos para comprender los términos que se emplean y las actividades en las que participarán para dar a conocer lo que realmente necesitan. El equipo de desarrollo muchas veces no presta el interés y el tiempo requerido en las etapas tempranas de desarrollo, a pesar de ser fundamentales y definitorias en la elaboración de todo producto de software, cuya realización ineficiente e inadecuada, influye negativamente en la satisfacción de las necesidades del cliente, en la entrega en tiempo y en la calidad de los productos, así como en los presupuestos destinados al desarrollo de los mismos.

Frederick P. Brooks comenta sobre la importancia de una correcta Ingeniería de requisitos: *“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 requerimientos del producto”* (Brooks, 1987).



## INTRODUCCIÓN

Es de vital importancia para un proyecto de software conocer a través de qué condiciones fue generado un requisito, qué le provee soporte o quién lo implementa para poder evaluar el impacto que tenga en el sistema cualquier cambio que sobre este se genere. En dependencia de las características presentes en cada uno de estos proyectos existen procesos que permiten, en mayor o menor medida, realizar un control sobre los requisitos.

El seguimiento a los requisitos en el Centro de Informatización para la Gestión de Entidades (CEIGE) se hace a nivel de línea de desarrollo. El analista principal de cada línea es el máximo responsable de mantener actualizada la información relacionada con los requisitos de software.

Esta información es almacenada en un repositorio y es mantenida durante todo el ciclo de vida del proyecto. Esta tiene como propósito mantener información sobre los requisitos, sus atributos y dependencias para ser usados en la evaluación del impacto y gestión de los cambios.

En el CEIGE, el registro de los cambios en los requisitos se realiza de forma manual. El analista principal tiene que visitar los puestos de trabajo de los analistas en cada uno de los diferentes módulos. Esto implica que se tengan que revisar los documentos que contienen esta información para actualizarlos posteriormente.

Para obtener información en relación al estado de los requisitos y la fase en que estos se encuentran es necesario escribir fórmulas en hojas de cálculo, pero de esta forma no es posible definir qué otros artefactos se ven afectados cuando un requisito cambia, por lo que la identificación de los riesgos y del impacto de los cambios de los requisitos en el proceso de desarrollo no es óptimo.

Por otro lado, se debe tener en cuenta que el equipo de desarrollo del CEIGE es muy numeroso y los módulos están muy interrelacionados, de manera que los cambios en los requisitos necesitan ser difundidos dentro de un amplio equipo, pues son de interés colectivo.

Los analistas para transmitir esta información en la actualidad, necesitan reunirse sistemáticamente en largas reuniones de integración para informar los cambios que se generan en los requisitos en los diferentes módulos. Estas reuniones ocupan el tiempo de los implicados, lo cual incide en una disminución de su tiempo de productividad.

## INTRODUCCIÓN

Debido a las razones anteriormente expuestas, en el CEIGE se definieron un grupo de actividades para realizar el seguimiento a los requisitos pero estas estaban pobremente descritas y fundamentadas, no abarcaban todos los pasos que exige el seguimiento a los requisitos. Además, estas actividades no eran cumplidas con la seriedad requerida porque se dejaban para posteriores momentos en el desarrollo de software en lugar de realizarlas desde el comienzo de la etapa de análisis de los proyectos.

Tras analizar el modo en que se realiza el seguimiento a los requisitos en el CEIGE, se concluye que es necesario que este proceso sea correctamente definido y establecido, que permita realizar eficientemente la identificación y el registro de los elementos de seguimiento y así conocer de forma óptima la trazabilidad de los requisitos de software.

A partir de la problemática expuesta en el presente trabajo de diploma, se plantea como **problema a resolver**: la imposibilidad de mantener el seguimiento a los requisitos de software en el CEIGE, lo que impide conocer la relación entre los requisitos identificados y de estos con otros elementos de software durante el proceso de desarrollo.

Consecuentemente, el **objeto de estudio** es la Gestión de Requisitos en el CEIGE, enmarcando el **campo de acción** en el Seguimiento a los Requisitos en dicho centro.

A partir del problema mencionado anteriormente el **objetivo general** es diseñar un procedimiento para mantener el seguimiento a los requisitos de software que permita gestionar las relaciones entre los elementos de seguimiento de software en el CEIGE.

Para dar cumplimiento al objetivo planteado se han trazado las siguientes tareas:

- Elaborar el diseño teórico de la investigación.
- Seleccionar una herramienta informática que permita realizar el seguimiento a los requisitos y justificar la elección.
- Proponer un procedimiento que permita mantener el seguimiento a los requisitos de software.
- Realizar una prueba piloto del procedimiento propuesto en una de las líneas del CEIGE.
- Validar el procedimiento propuesto a partir de los resultados obtenidos durante la prueba piloto.

## INTRODUCCIÓN

Se formula como **idea a defender** que un procedimiento para el seguimiento a los requisitos enfocado a las necesidades del CEIGE permitirá conocer las relaciones entre los requisitos con otros elementos de software durante el proceso de desarrollo.

# Capítulo 1. FUNDAMENTACIÓN TEÓRICA

## 1.1. INTRODUCCIÓN

En el presente capítulo se desarrollará el marco teórico de la investigación, incluyendo además la fundamentación teórica sobre la Ingeniería de requisitos y Gestión de requisitos, centrándose en sus características, importancia, haciendo mayor énfasis en el Seguimiento de los requisitos de software.

## 1.2. PROCESO DE DESARROLLO DE SOFTWARE

El desarrollo de software requiere de la aplicación de un proceso de desarrollo que permita garantizar la calidad de los productos generados y que satisfaga las necesidades de las personas que lo usarán.

La palabra proceso tiene su origen en el término latino *processus*: acción o sucesión de acciones continuas regulares, que ocurren o se llevan a cabo de una forma definida, y que llevan al cumplimiento de algún resultado; una operación continua o una serie de operaciones (Real Academia Española, 2010).

La serie de Normas de Calidad ISO 9003:2000 define un proceso como: “Conjunto de actividades mutuamente relacionadas o que interactúan, las cuales transforman elementos de entrada en salidas” (Normalización, 2006) El software como producto también sigue un proceso de producción o desarrollo. El proceso de desarrollo de software es una descripción de la construcción del software, que contiene actividades organizadas de modo que en conjunto producen código probado. No existe una definición estándar de estas actividades y muchos autores le dan importancia a algunas más que a otras.

Un proceso define quién está haciendo qué, cuándo, y cómo para alcanzar un determinado objetivo [...]. Es necesario que el proceso sirva como guía para todos los participantes –clientes, usuarios, desarrolladores y directores ejecutivos (Jacobson, 2000).

## 1.3. REQUISITO DE SOFTWARE

La IEEE (Institute of Electrical and Electronics Engineers) ha definido el concepto de requisito como:

*“i) Una condición o capacidad que necesita el usuario para resolver algún problema o alcanzar un objetivo.*

*ii) Condición o capacidad que debe cumplir o poseer un sistema o componente del sistema para satisfacer un contrato, estándar, especificación u otro documento formalmente impuesto.*

*iii) Una representación documental de una condición o capacidad como en i o en ii.” (IEEE, 1998)*

Un conjunto de requisitos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo. Necesario, conciso, completo, consistente, no ambiguo y verificable son las características de un requisito y a la vez sus principales propiedades (Brackett, 1990).

Un requisito es necesario si su omisión provoca una deficiencia en el sistema a construir y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso. Es conciso si es fácil de leer y entender. Está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión. Es consistente si no es contradictorio con otro requisito. Un requisito no es ambiguo cuando tiene una sola interpretación, o sea, el lenguaje usado en su definición no debe causar confusiones al lector. Un requisito es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o pruebas (Brackett, 1990).

Cuando se pretende especificar los requisitos frecuentemente se halla una serie de dificultades que unidas a las características que deben tener hacen más difícil la extracción de éstos. Por ejemplo, los requisitos son obvios y vienen de muchas fuentes por lo que son difíciles de expresar en palabras. Existen muchos tipos de requisitos, de hardware, de software, de seguridad, por lo que su cantidad en un proyecto puede ser difícil de manejar. Nunca son iguales, algunos son más riesgosos, más importantes o más estables que otros. Están relacionados unos con otros y a su vez se relacionan con otros artefactos del proceso de desarrollo. Cada uno tiene propiedades únicas y abarcan áreas funcionales específicas. Los requisitos pueden cambiar a lo largo del ciclo de desarrollo.

### 1.3.1. REQUISITOS FUNCIONALES Y REQUISITOS NO FUNCIONALES

Los requisitos de software están divididos en funcionales y no funcionales, seguidamente se muestran algunas de las características específicas de cada uno de estos tipos de requisitos.

Los requisitos funcionales son parte de los requerimientos del sistema e indican características y restricciones sobre la funcionalidad del software.

Los requisitos no funcionales definen propiedades y restricciones del sistema. Se refiere principalmente a la fiabilidad, el tiempo de respuesta, la capacidad de almacenamiento, la capacidad de los dispositivos de entrada/salida, y la representación de datos que se utiliza en las interfaces del sistema.

Los requisitos no funcionales forman una parte importante en el logro de la calidad de los productos de software, principalmente para que clientes y usuarios puedan valorar las características no funcionales del producto. Cumpliendo con toda la funcionalidad requerida entonces las propiedades no funcionales, como cuán usable, seguro, conveniente y agradable, pueden marcar la diferencia entre un producto bien aceptado y uno con poca aceptación.

### 1.4. INGENIERÍA DE REQUISITOS

La Ingeniería de requisitos es el proceso de descubrimiento y comunicación de las necesidades de clientes y usuarios. Uno de los aspectos más importantes de ella es la comunicación, característica ésta que vuelve al proceso complejo por la alta presencia del factor humano que contiene y es el responsable de que la disciplina contenga aspectos sociales y culturales y no sólo de índole técnica. Además, la Ingeniería de requisitos debe ser considerada como un proceso de construcción de una especificación de requisitos en el que se avanza desde especificaciones iniciales, que no poseen las características oportunas, hasta especificaciones finales completas, formales y acordadas entre todas las partes.

Roger Pressman define la Ingeniería de requisitos como *“un proceso de descubrimiento, refinamiento, modelado y especificación. Se refinan en detalle los requisitos del sistema y el papel asignado al software.”* (Pressman, 2002)

Para Thayer y Dorfman *“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 la solución sin ambigüedad, validando la especificación y gestionando los requisitos para que se transformen en un sistema operacional.” (Thayer, 1997)*

En tanto, Donald Reifer ha plasmado que *“la ingeniería de requisitos es el uso sistemático de procedimientos, técnicas, lenguajes y herramientas para obtener con un coste reducido el análisis, documentación, evolución continua de las necesidades del usuario y la especificación del comportamiento externo de un sistema que satisfaga las necesidades del usuario.” (Reifer, 1994)*

### 1.4.1. ACTIVIDADES EN LA INGENIERÍA DE REQUISITOS

Diferentes autores descomponen al proceso de Ingeniería de requisitos en diversas actividades. Todos están de acuerdo en cuanto al objetivo que guía cada una de estas actividades. Lo anterior se puede resumir en tres objetivos fundamentales:

- elicitar los requisitos de las diversas fuentes individuales;
- asegurar que las necesidades de todos los usuarios son consistentes y factibles; y
- validar que los requisitos que se derivaron son un reflejo exacto de las necesidades del usuario.

Según Ian Sommerville y Pete Sawyer estas actividades son (Sommerville, 2002):

- Identificación de requisitos.
- Análisis y Negociación de requisitos.
- Especificación de requisitos.
- Validación de requisitos.
- Gestión de requisitos.

La Identificación de requisitos consiste en interactuar con el cliente. Es una comunicación iterativa con los clientes y los interesados del proyecto, para descubrir, definir, refinar y registrar una representación precisa de los requisitos del futuro sistema.

Durante el Análisis y Negociación de los requisitos, estos son agrupados por categorías y se organizan en subconjuntos; al mismo tiempo, son examinados en su consistencia, completitud y ambigüedad y se clasifican en base a las necesidades de los clientes y usuarios.

La Especificación de requisitos constituirá un convenio formal con el cliente y un compromiso para el equipo de desarrollo de construir exactamente lo que ha solicitado el cliente.

El siguiente paso del proceso es la Validación de requisitos, que mediante revisiones técnicas formales, se examinan las especificaciones para asegurar que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin inconsistencias y sin omisiones. Tras este proceso de verificación se comprueba que los errores detectados han sido corregidos y que el resultado del trabajo se ajusta a los estándares establecidos para el proyecto.

Una correcta Ingeniería de requisitos permite gestionar las necesidades del proyecto en forma estructurada ya que cada una de sus actividades consiste de una serie de pasos organizados y bien definidos. Disminuye los costos y retrasos del proyecto, se ha mostrado que reparar errores por un mal desarrollo no descubierto a tiempo, es sumamente caro. La Ingeniería de requisitos obliga al cliente a considerar sus requisitos cuidadosamente y revisarlos dentro del marco del problema, se involucra durante todo el desarrollo del proyecto, por lo que evita rechazos de usuarios finales.

### 1.4.2. CARACTERÍSTICAS DE LA INGENIERÍA DE REQUISITOS

Las bibliografías consultadas referentes a la Ingeniería de requisitos están de acuerdo con las siguientes características (Chaves, 1997):

- La Ingeniería de requisitos es un proceso iterativo, ya que al ser un proceso de descubrimiento y comunicación, difícilmente llegará a realizarse en forma lineal.
- Los requisitos no siempre son entregados en su totalidad por los clientes y usuarios, así que los ingenieros de requisitos también deben saber descubrirlos.
- Los límites de las actividades de la Ingeniería de requisitos son difíciles de establecer por la misma naturaleza del proceso.
- No hay claridad ni consenso en cuanto a los productos que se deben tener al final del flujo de trabajo. Los requisitos pueden evolucionar tan rápidamente que pueden cambiar antes de haber concluido el desarrollo del sistema.



### 1.4.3. IMPORTANCIA DE LA INGENIERÍA DE REQUISITOS.

Los principales beneficios que se obtienen de la Ingeniería de requisitos son (Chaves, 1997):

- Permite gestionar las necesidades del proyecto en forma estructurada: Cada actividad de la Ingeniería de requisitos consiste de una serie de pasos organizados y bien definidos.
- Mejora la capacidad de predecir cronogramas de proyectos, así como sus resultados: La Ingeniería de requisitos proporciona un punto de partida para controles subsecuentes y actividades de mantenimiento, tales como estimación de costos, tiempo y recursos necesarios.
- Disminuye los costos y retrasos del proyecto: Muchos estudios han demostrado que reparar errores por un mal desarrollo no descubierto a tiempo, es sumamente caro.
- Mejora la calidad del software: La calidad en el software tiene que ver con cumplir un conjunto de requisitos (funcionalidad, facilidad de uso, confiabilidad, desempeño, etc.).
- Mejora la comunicación entre equipos: La especificación de requisitos representa una forma de consenso entre clientes y desarrolladores. Si este consenso no ocurre, el proyecto no será exitoso.
- Evita rechazos de usuarios finales: La Ingeniería de requisitos obliga al cliente a considerar sus requisitos cuidadosamente y revisarlos dentro del marco del problema, por lo que se le involucra durante todo el desarrollo del proyecto.

### 1.5. GESTIÓN DE REQUISITOS

La Gestión de requisitos se identifica actualmente como una muy buena práctica que contribuye, como ninguna otra, al éxito de los proyectos de software, al posibilitar un entendimiento común entre el cliente y el grupo de desarrolladores de los requisitos del cliente que deben concebirse en el producto final, la comprensión de los problemas que se necesitan solucionar y las posibles vías de resolverlos (Agusti, 2008).

En la actualidad persisten problemas en el desarrollo de software, entre ellos, un inadecuado entendimiento de las necesidades de los usuarios, incapacidad de absorber cambios en los requisitos e insatisfacciones de los clientes por inaceptable o bajo desempeño del software. Las principales causas de

estos hechos son la administración insuficiente de requisitos; los problemas que afectan la comunicación; las inconsistencias no detectadas entre requisitos, diseño y programación; las validaciones tardías de requisitos; el enfrentamiento reactivo de riesgos y la propagación de cambios sin control lo cual persiste a lo largo de la vida del sistema. Por lo tanto, es esencial planear posibles cambios a los requisitos cuando el sistema sea desarrollado y utilizado. Por tanto, la Gestión de requisitos adquiere una importancia vital dentro de la Ingeniería de requisitos.

Roger Pressman describe la Gestión de requisitos como *“un conjunto de actividades que ayudan al equipo de desarrollo a identificar, controlar y darle seguimiento a los requisitos y los cambios en cualquier momento”*. (Pressman, 2002)

Bárbara McDonald Landazuri afirma que la gestión de requisitos es un componente esencial en el desarrollo de un proyecto de software ya que provee la dirección y alcance del proyecto. Expone además, que la gestión de requisitos es el proceso encargado de la identificación, asignación y seguimiento de los requisitos, incluyendo la verificación, la modificación y el control de su estado a lo largo del ciclo de vida del proyecto.

A partir de las definiciones anteriores se puede resumir que la Gestión de requisitos es el conjunto de actividades que ayudan al equipo de trabajo a identificar, controlar y seguir los requisitos y sus cambios en cualquier momento.

### **1.6. SEGUIMIENTO DE LOS REQUISITOS**

Específicamente, el seguimiento a los requisitos está sustentado en el concepto de trazabilidad de los requisitos que algunos autores han ofrecido.

Para Francisco Pinheiro, *“la trazabilidad es la habilidad de definir, capturar y seguir las trazas dejadas por los requerimientos en otros elementos del ambiente de desarrollo de software, así como las trazas que estos elementos dejan en los requisitos”*. (Pinheiro, 2000)

Asimismo, la definición de trazabilidad, que se reconoce como la más difundida, *“se concentra en la especificación de requisitos de software y requiere en primer lugar, que permita remontarse hacia atrás en*

*el origen de los requerimientos y que también posibiliten referenciar a cada requerimiento en la documentación de desarrollo". (IEEE, 1998)*

A partir de estas definiciones se puede entender el proceso de seguimiento de los requisitos como una actividad que posibilitará mostrar la relación existente entre los requisitos identificados y de estos con otros artefactos del sistema. Esto supone un amplio beneficio, en tanto ayudará a realizar un análisis del impacto de los cambios, lo que favorecerá la reducción de los riesgos y paralelamente será posible demostrar que se implementaron todos los requisitos.

### 1.6.1. MATRICES DE SEGUIMIENTO

Una vez los requisitos han sido identificados, se desarrollarán un conjunto de matrices para su seguimiento. Cada matriz de seguimiento identifica los requisitos relacionados con uno o más aspectos del sistema o su entorno. Entre las posibles matrices de seguimiento se encuentran (Pressman, 2002):

- Matriz de seguimiento de características: Muestra los requisitos identificados en relación a las características definidas por el cliente del sistema-producto.
- Matriz de seguimiento de orígenes: Identifica el origen de cada requisito.
- Matriz de seguimiento de dependencias: Indica cómo se relacionan los requisitos entre sí.
- Matriz de seguimiento de subsistemas: Vincula los requisitos a los subsistemas que los manejan.
- Matriz de seguimiento de interfaces: Muestra cómo los requisitos están vinculados a las interfaces externas o internas del sistema.

## 1.7. HERRAMIENTAS

Existen varias herramientas capaces de facilitar, en lo posible, la gestión de los requisitos. Estas herramientas son, en mayor o menor medida complejas y sofisticadas, pero ahorran en costos y minimizan el impacto de errores.

A continuación se presenta un análisis sobre algunas de las herramientas capaces de mantener una gestión de requisitos basado principalmente en la funcionalidad que estas presentan para seguir la trazabilidad de los requisitos a lo largo de todo el ciclo de vida del proyecto.

### 1.7.1. GATHERSPACE.

Es una herramienta que gestiona toda la fase de análisis de un proyecto mediante una aplicación web, esto trae como inconveniente que su manejo puede ser lento puesto que se está a expensas de un servidor (Pescador, 2009).

Es una herramienta web sencilla y por ello es muy limitada en cuanto a funcionalidades que son útiles cuando se realiza un proyecto completo y bien documentado. No dispone de control de versionado, ni historial, ni se pueden crear líneas base, lo que hace que la información del proyecto esté incompleta respecto a los cambios producidos en el mismo (Pescador, 2009).

### 1.7.2. SOFTWARE PLANNER 9.0.

Es una herramienta que está más orientada a la gestión de proyectos que a la gestión de requisitos, debido a esto no cuenta con las funcionalidades requeridas (Pescador, 2009).

### 1.7.3. REM (REQUIREMENTS MANAGEMENT).

REM es una aplicación de gestión de requisitos desarrollada por la Universidad de Sevilla, por tanto, no cuenta con las funcionalidades requeridas para los proyectos del CEIGE. Se trata de un proyecto libre, no comercial, y con menos opciones que las otras herramientas del estudio. Genera reportes en ficheros HTML que resultan ser poco flexibles y muy estrictos en cuanto a su forma. Es una herramienta destinada a proyectos pequeños (Hidalgo, 2009).

### 1.7.4. OSRMT (OPEN SOURCE REQUIREMENTS MANAGEMENT TOOL).

Dentro de las herramientas de código abierto que abarca la Ingeniería de requisitos, se trata de la herramienta que mejores funcionalidades ofrece por esto es una de las propuestas del proceso de mejoras de la UCI para realizar el seguimiento a los requisitos pero estas funcionalidades son escasas debido a su escasa madurez. También presenta problemas con la usabilidad, puesto que su diseño gráfico resulta incómodo al realizar algunas de sus funcionalidades. Además, el versionado de la herramienta no está organizado y las sucesivas versiones no son compatibles entre sí (Pescador, 2009).

### 1.7.5. REQUISITEPRO

RequisitePro permite configurar y dar seguimiento a las relaciones entre requisitos. Crea una línea base basada en XML usado para armar rápidamente nuevos proyectos o para comparar con otras líneas base del proyecto, exponiendo los cambios y omisiones a los requisitos en múltiples niveles de detalle. Permite crear y exportar matrices de trazabilidad y reportes de atributos de los requisitos. A todos los artefactos se les denomina requisitos, lo cual es impreciso cuando se relacionan distintos tipos de elementos (Vázquez, 2007).

### 1.7.6. ENTERPRISE ARCHITECT (EA)

EA es una herramienta de uso muy sencillo, que cubre el desarrollo de software desde la captura de requisitos a lo largo de las etapas de análisis, diseño, pruebas y mantenimiento. Permite generar documentación e informes flexibles y de alta calidad. Permite crear una línea base basada en XML. Proporciona trazabilidad completa desde el análisis de requisitos y los artefactos de diseño, hasta la implementación y el despliegue. El Lenguaje de Modelamiento Unificado (UML, por sus siglas en inglés) ofrece beneficios significativos para ayudar a construir modelos de software rigurosos, donde es posible mantener la trazabilidad de los requisitos (Pescador, 2009).

Luego de realizar un estudio sobre las herramientas existentes para el seguimiento de los requisitos se concluye que:

Existe una amplia variedad de herramientas para el seguimiento a los requisitos y todas con diversas características, además, muchas se encuentran en prueba por lo que no son confiables y cuentan con pocas funcionalidades. Otras, como OSRMT, se presentan como software libre pero no cuentan con las funcionalidades requeridas para ser utilizadas en los proyectos del CEIGE, además presenta problemas con el versionado y la usabilidad.

Otras herramientas como EA y RequisitePro cuentan con las funcionalidades requeridas para realizar correctamente el seguimiento a los requisitos en el centro. De estas herramientas se propone EA porque, a diferencia de RequisitePro, permite realizar conjuntamente el seguimiento a los requisitos y el modelado de las demás actividades del ciclo de vida de un proyecto de software, simplificando para el centro la realización de estas actividades.

### **1.8. CONCLUSIONES PARCIALES**

Los problemas en los requisitos son una de las principales causas por las cuales los proyectos de software no cumplen los cronogramas, sobrepasen los costos y que se obtengan sistemas que no satisfacen las expectativas de los clientes y usuarios finales.

Considerando los elementos señalados anteriormente, es necesario definir un procedimiento para desarrollar el Seguimiento de los requisitos en el CEIGE que se adecúe específicamente a las características de este, usando Enterprise Architect como herramienta de gestión de requisitos.

## Capítulo 2. PROPUESTA DE SOLUCIÓN

### 2.1 INTRODUCCIÓN

En este capítulo se presenta la propuesta de solución del problema planteado en la Introducción del presente trabajo de diploma, que radica en la creación de un procedimiento para el seguimiento a los requisitos de software en el CEIGE que mejore la calidad del proceso de seguimiento. Se especificarán detalladamente los roles que intervienen en cada actividad propuesta y los artefactos generados en estas.

### 2.2 CARACTERÍSTICAS DEL PROCEDIMIENTO

El procedimiento está conformado por un conjunto de actividades organizadas lógicamente y secuencialmente que posibilita realizar el seguimiento a los requisitos a través de todo el proceso de desarrollo de software. Permite seguir las trazas del requisito desde su identificación hasta la implementación mediante el uso de la herramienta seleccionada en la Fundamentación Teórica del presente trabajo de diploma.

### 2.3 ROLES

En la siguiente tabla se listan los roles propuestos para que formen parte en el proceso de seguimiento a los requisitos.

**Tabla 1. Roles**

Roles	Descripción
<b>Analista</b>	Responsable por parte del equipo de desarrollo de la identificación y extracción de los requisitos, además de la especificación de estos. Debe validar junto al cliente los requisitos y controlar los cambios que se producen en los requisitos de su negocio.
<b>Analista Principal</b>	Define los lineamientos para desarrollar el proceso de seguimiento a los requisitos. Lleva un control global del comportamiento de los requisitos en el

	equipo de desarrollo. Encargado de analizar posible reutilización, integración, etc.
<b>Comité de control de cambios</b>	Encargado por parte del equipo de desarrollo de recibir, estudiar y aprobar las solicitudes de cambio que le son presentadas.

## 2.4 ROLES POR ACTIVIDADES

En el siguiente diagrama se describe la participación de cada rol en el proceso de seguimiento a los requisitos.



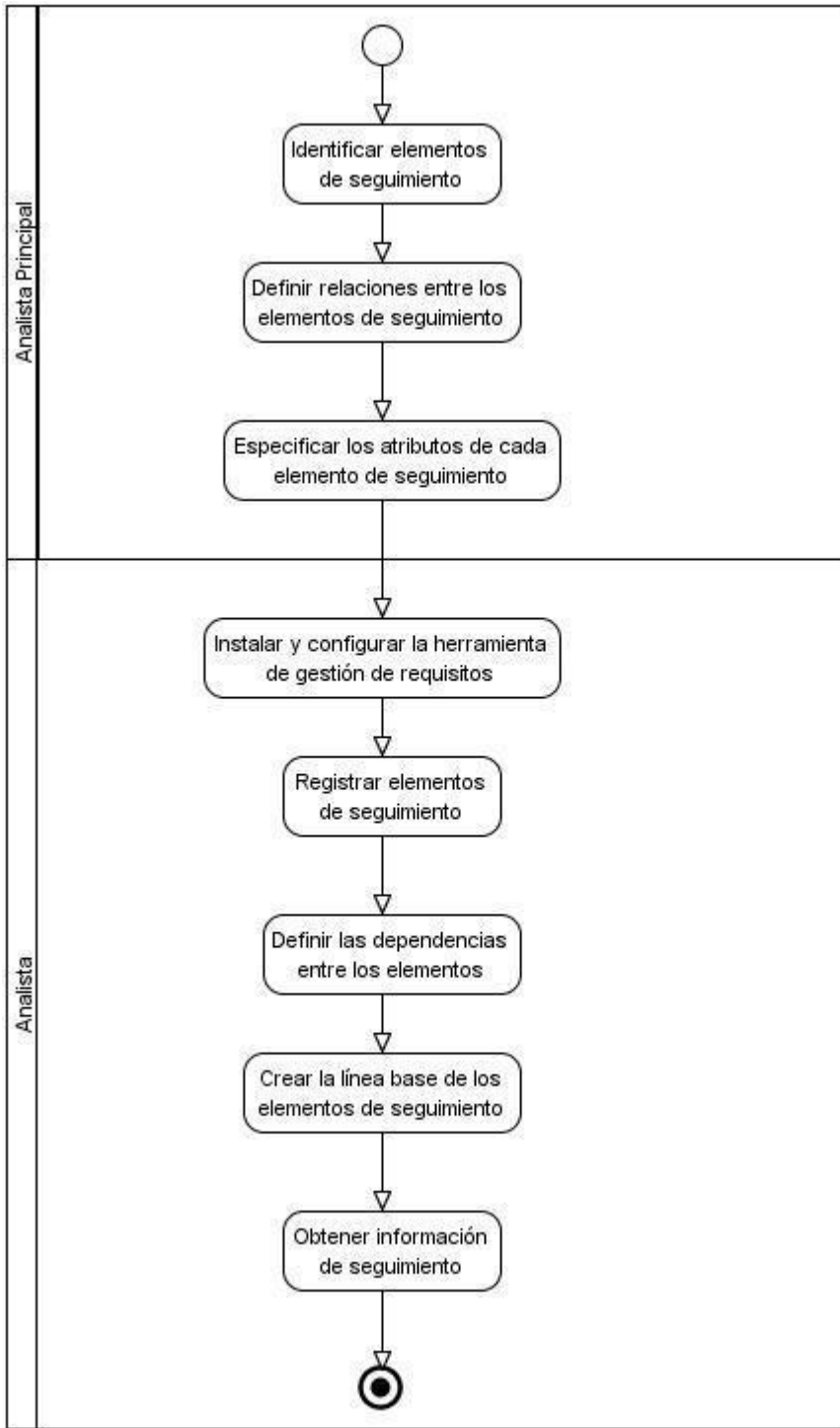


Figura 1. Roles por actividades.

## 2.5 PROCEDIMIENTO PARA EL SEGUIMIENTO A LOS REQUISITOS. ESTRUCTURA

Con el objetivo de lograr una mejor organización en la realización del seguimiento a los requisitos en el CEIGE se ha decidido dividir el procedimiento en ocho pasos fundamentales siguiendo las características y necesidades existentes en el centro. Esta secuencia de pasos será explicada con mayor precisión en el desarrollo del presente capítulo.

1. Identificar elementos de seguimiento.
2. Definir relaciones entre los elementos de seguimiento a los requisitos.
3. Especificar los atributos de cada elemento de seguimiento.
4. Instalar y configurar la herramienta de gestión de requisitos.
5. Registrar elementos de seguimiento a los requisitos.
6. Definir las dependencias entre los elementos de seguimiento.
7. Crear la línea base de los elementos de seguimiento.
8. Obtener información de seguimiento.

### 2.5.1 PASO 1. IDENTIFICAR ELEMENTOS DE SEGUIMIENTO.

Este paso consiste en determinar cuáles son los elementos necesarios para realizar el seguimiento a los requisitos, o sea, los que están relacionados directamente con los requisitos del sistema, por ejemplo, los elementos que dan soporte a los requisitos o los encargados de implementarlos.

Se propone realizar esta actividad en reuniones entre el equipo de analistas y el analista principal para llegar a un consenso en cuanto a la selección de los elementos ya que estos son únicos para cada proyecto de software en dependencia de las características y necesidades del equipo de desarrollo.

**Tabla 2. Paso 1. Identificar elementos de seguimiento.**

Paso 1	
Nombre	Identificar elementos de seguimiento.

<b>Objetivos</b>	Identificar los elementos de seguimiento acorde a las necesidades del proyecto.
<b>Roles</b>	Analista principal.
<b>Otros participantes</b>	Analistas.
<b>Artefactos</b>	Guía para el seguimiento a los requisitos.

Los elementos de seguimiento identificados para el CEIGE se encuentran descritos en la Guía para el seguimiento a los requisitos, estos son mostrados en la siguiente tabla.

**Tabla 3. Elementos de seguimiento.**

Artefacto	Elemento de seguimiento	Descripción
Proyecto técnico	* Necesidad de la organización	Este elemento encierra tanto las necesidades de los involucrados como las características básicas u objetivos del sistema.
Descripción de proceso de negocio	* Proceso de negocio	Representa los procesos de negocio que deben ser apoyados por la solución informática que se desarrolla.
Especificación de requisitos de software	* Agrupación de requisitos	Representa conjuntos de requisitos relacionados por sus objetivos, Por ejemplo: el conjunto de requisitos para gestionar un concepto o el conjunto de requisitos que cambian de estado a un concepto.

Descripción de requisitos	* Requisito funcional	Se refiere tanto a los requisitos funcionales como a las salidas del sistema también conocidas como reportes.
Salida del sistema		
Especificación de requisitos de software	* Requisito no funcional	Se refiere a los requisitos de sistema, atributos de calidad, requisitos de interfaz y restricciones.
Documento de arquitectura del sistema	* Componente	Representa cada componente definido en la arquitectura de sistema.
Registro de revisiones y cambios	Solicitud de cambio	Se refiere a la solicitud de modificaciones, adición o eliminación de un requisito de la línea base.

### 2.5.2 PASO 2. DEFINIR LAS RELACIONES ENTRE LOS ELEMENTOS DE SEGUIMIENTO

Este paso consiste en establecer las relaciones entre los elementos de seguimiento, o sea, definir el tipo de relación que existe entre estos elementos.

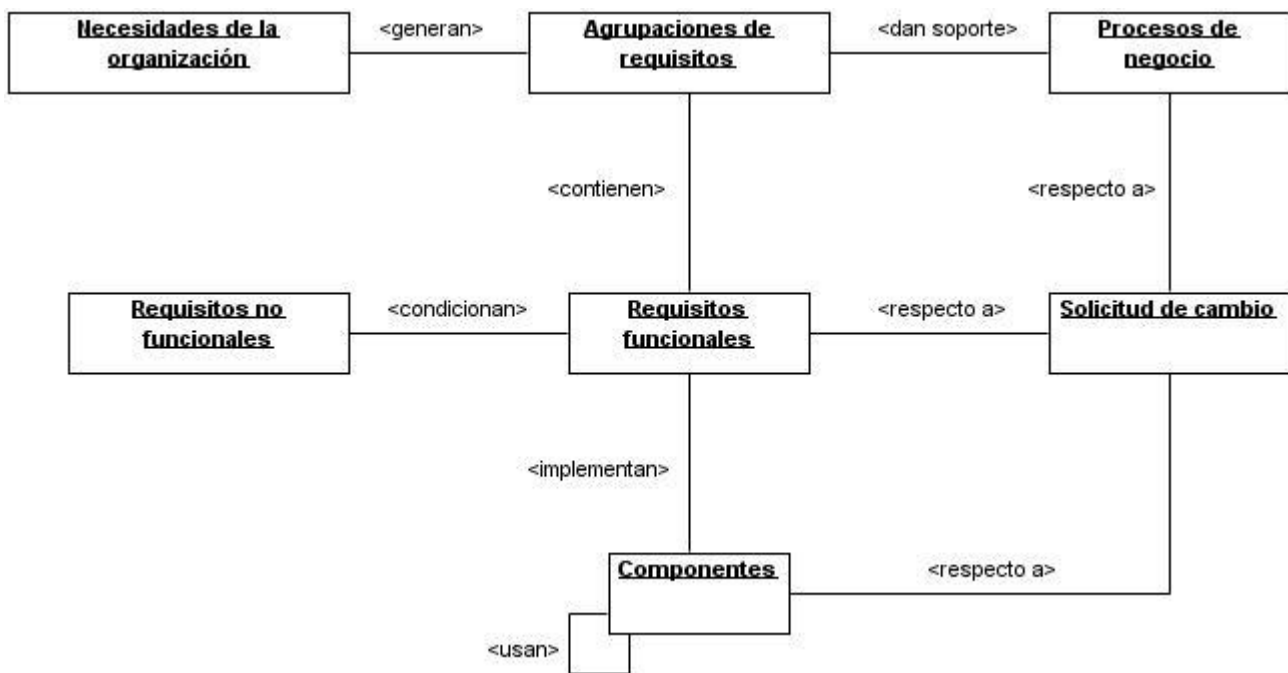
En dependencia del tipo de elemento es la conexión que se establece entre estos. Esta actividad puede ser realizada por el equipo de analistas paralelamente a la identificación de los elementos.

**Tabla 4. Paso 2. Definir las relaciones entre los elementos de seguimiento.**

Paso 2	
Nombre	Definir las relaciones entre los elementos de seguimiento.

<b>Objetivos</b>	Establecer el tipo de relaciones que existen entre los elementos.
<b>Roles</b>	Analista principal.
<b>Otros participantes</b>	Analistas.
<b>Artefactos</b>	Guía para el seguimiento a los requisitos.

Las relaciones definidas para los elementos de seguimiento identificados en el Paso 1 están especificadas en la Guía para el seguimiento a los requisitos y se muestran en la siguiente figura.



**Figura 2. Relaciones entre los elementos de seguimiento.**

El seguimiento a los requisitos debe ser bidireccional, al relacionar un elemento se indica su origen y a cuál él genera. Las conexiones definidas anteriormente se describen a continuación.

- Relación Necesidad de la organización-Agrupación de requisitos: las agrupaciones de requisitos se derivan de las necesidades de la organización. Debe conocerse para una necesidad de la organización qué agrupaciones la satisfacen y para cada agrupación debe conocerse a cuáles necesidades ella satisface.
- Relación Proceso de negocio-Agrupación de requisitos: las agrupaciones de requisitos dan soporte a procesos de negocio específico. Debe conocerse qué agrupaciones apoyan a cada proceso de negocio y cada proceso de negocio de qué agrupación recibe apoyo.
- Relación Agrupación de requisitos-Requisito funcional: esta relación no se representa mediante el uso del seguimiento a los requisitos, las agrupaciones son conjuntos de requisitos relacionados, por tanto, una agrupación de requisitos contiene un grupo de requisitos. Esto quiere decir que no debe mantenerse una relación de seguimiento entre las agrupaciones y los requisitos funcionales.
- Relación Requisito no funcional-Requisito funcional: los requisitos no funcionales condicionan los requisitos funcionales. Debe conocerse cada requisito no funcional a qué requisitos funcionales condicionan y cada requisito funcional por cuáles no funcionales es afectado.
- Relación Requisito funcional-Componente: para que el software desarrollado satisfaga los requisitos funcionales impuestos deben implementarse un grupo de componentes según determinados estándares, restricciones o soluciones de diseño que garantizan el cumplimiento del requisito. Debe conocerse cada componente qué requisitos implementa y cuáles son los componentes que implementan a cada requisito.
- Relación Componente-Componente: en un software unos componentes usan a otros, esta relación debe mantenerse como parte del seguimiento para evaluar el impacto de los cambios. De cada componente debe conocerse de qué componentes depende y cuáles dependen de él.
- Relación Solicitud de cambio-Requisito funcional o Solicitud de cambio-Componente o Solicitud de cambio-Proceso de negocio: el objetivo de esta relación es conocer cada solicitud de cambio sobre qué elementos se realiza.

2.5.3 PASO 3. ESPECIFICAR LOS ATRIBUTOS DE CADA ELEMENTO DE SEGUIMIENTO

Una vez concluidos los pasos 1 y 2 se deben establecer los atributos para cada uno de los elementos de seguimiento identificados.

**Tabla 5. Paso 3. Especificar atributos de cada elemento de seguimiento.**

<b>Paso 3</b>	
<b>Nombre</b>	Especificar los atributos de cada elemento de seguimiento.
<b>Objetivos</b>	Definir los estados y posibles valores de cada elemento de seguimiento.
<b>Roles</b>	Analista principal.
<b>Otros participantes</b>	Analistas.
<b>Artefactos</b>	Guía para el seguimiento a los requisitos.

Los atributos son características, estados y posibles valores que puede tener un elemento de seguimiento. En dependencia del elemento son los atributos definidos pero generalmente para todos se registra identificador, prioridad, estado y referencia.

Los atributos definidos para los elementos de seguimiento identificados en el CEIGE están descritos en la Guía para el seguimiento a los requisitos. En la secuencia de tablas que se muestran a continuación están descritos los atributos por elementos.

**2.5.3.1 Necesidad de la organización**

**Tabla 6. Necesidades de la organización.**

<b>Atributo</b>	<b>Descripción</b>	<b>Posibles valores</b>	<b>Descripción</b>
-----------------	--------------------	-------------------------	--------------------

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

Id	Identifica de manera única al elemento de seguimiento.	N/A	N/A
Estado	Establece en qué estado se encuentra la necesidad de los interesados	Identificada	Cuando se identifica una necesidad esta se registra, con una breve descripción, en el estado: identificada.
		Descrita	Una necesidad pasa a este estado cuando se han identificado y descrito los requisitos que la satisfacen.
		Resuelta	Cuando se ha validado y verificado por todos los interesados que el sistema informático satisface la necesidad, esta pasa al estado: resuelta.
Prioridad	Indica la prioridad de la necesidad.	Pendiente	Necesidad que requiere revisión para determinar su prioridad.
		No requerida	Es una necesidad que no es obligatorio satisfacer en la actual versión del sistema.



## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

		Importante	La necesidad debe ser implementada en la actual iteración si es posible.
		Obligatoria	La necesidad tiene que satisfacerse en la presente versión del sistema.
Breve descripción	Describe brevemente la necesidad, en términos de los problemas que resuelve.	N/A	N/A
Versión objetivo	Establece en qué versión del sistema se debe resolver la necesidad.	N/A	N/A
Referencia	Indica la dirección en el repositorio de control de versiones del documento en el que está descrita la necesidad de la organización.	N/A	N/A

2.5.3.2 Proceso de negocio

Tabla 7. Proceso de negocio.

Atributo	Descripción	Posibles valores	Descripción
Id	Identifica de manera única al elemento de seguimiento.	N/A	N/A
Estado	Establece en qué estado se encuentra el proceso de negocio	Identificado	Cuando se identifica un proceso de negocio este se registra, con una breve descripción, en el estado identificado.
		Descrito	Un proceso de negocio pasa a este estado cuando se ha completado totalmente la plantilla de descripción de proceso de negocio del mismo.
		Aprobado	Un proceso de negocio pasa a este estado cuando se ha aprobado por parte de todos los interesados, en este momento pasa a formar parte de una línea base.
Prioridad	Indica la prioridad del	Alta	Los procesos de

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

	proceso de negocio.		negocio cuya descripción es crítica para el desarrollo del resto del proyecto tienen esta categoría. Es importante considerar la prioridad desde el punto de vista de negocio.
		Media	Los procesos de negocio que son necesarios pero no críticos para el desarrollo del proyecto tienen esta categoría.
		Baja	Los procesos de negocio cuya consideración en el proyecto son un valor añadido al alcance inicial.
Breve descripción	Describe brevemente al proceso de negocio en términos de sus objetivos, entradas y salidas principales.	N/A	N/A
Reusabilidad	Origen	Cuando existe un Modelo de Negocio que se está usando	N/A

	<p>como entrada al modelado se completa este atributo.</p> <p>El origen es el nombre del sistema del que proviene el modelo, por ejemplo: UCID, VERSAT.</p>		
%	<p>A partir de la evaluación del especialista funcional se establece el % de reusabilidad del proceso.</p>	0-100%	N/A
Versión objetivo	<p>Establece en qué versión del sistema se debe informatizar el proceso.</p>	N/A	N/A
Referencia	<p>Indica la dirección en el repositorio de control de versiones del documento en el que está descrito el proceso de negocio.</p>	N/A	N/A

### 2.5.3.3 Agrupación de requisitos

Tabla 8. Agrupación de requisitos.

Atributo	Descripción	Posibles valores	Descripción
----------	-------------	------------------	-------------

Id	Identifica de manera única al elemento de seguimiento.	N/A	N/A
Breve descripción	Describe brevemente al requisito en términos de lo que debe hacer el sistema.	N/A	N/A
Referencia	Indica la dirección en el repositorio de control de versiones del documento en el que está descrito el requisito.	N/A	N/A

#### 2.5.3.4 Requisito funcional

Tabla 9. Requisitos funcionales.

Atributo	Descripción	Posibles valores	Descripción
Id	Identifica de manera única al elemento de seguimiento.	N/A	N/A
Estado	Establece en qué estado se encuentra el requisito.	Identificado	Cuando se identifica un requisito este se registra, con una breve descripción, en el estado identificado.
		Descrito	Un requisito pasa a este estado cuando se ha

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

---

completado totalmente la plantilla de descripción requisito del mismo.

---

Aprobado

Un requisito pasa a este estado cuando se ha aprobado por parte de todos los interesados, en este momento pasa a formar parte de una línea base.

---

Rechazado

Un requisito es rechazado cuando no cumple al menos uno de los criterios para validar los requisitos.

---

Prioridad

Indica la prioridad del requisito. Alta

Tienen prioridad alta los requisitos que apoyan procesos priorizados como de prioridad alta, resuelven necesidades de los interesados de prioridad alta o son críticos para la arquitectura.

---

Media

Tienen prioridad media los requisitos que apoyan procesos

---

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

				priorizados como de prioridad media y resuelven necesidades de los interesados de prioridad media.
			Baja	Son los requisitos cuya implementación constituye un valor añadido al alcance previsto.
Breve descripción		Describe brevemente al requisito en términos de lo que debe hacer el sistema.	N/A	N/A
Reusabilidad	Origen	<p>Cuando existe una Especificación de Requisitos de Software que se está usando como entrada al modelado se completa este atributo.</p> <p>El origen es el nombre del sistema del que proviene el modelo, por ejemplo: UCID, VERSAT.</p>	N/A	N/A
	%	A partir de la evaluación del	0-100%	N/A

	especialista funcional se establece el % de reusabilidad del requisito.		
Versión objetivo	Establece en qué versión del sistema se debe implementar el requisito.	N/A	N/A
Referencia	Indica la dirección en el repositorio de control de versiones del documento en el que está descrito el requisito.	N/A	N/A

### 2.5.3.5 Requisito no funcional

Tabla 10. Requisitos no funcionales.

Atributo	Descripción	Posibles valores	Descripción
Id	Identifica de manera única al elemento de seguimiento.	N/A	N/A
Estado	Establece en qué estado se encuentra el requisito.	Identificado	Cuando se identifica un requisito este se registra, con una breve descripción, en el estado identificado.
		Descrito	Un requisito pasa a este estado cuando se ha



## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

---

		completado totalmente la plantilla de descripción requisito del mismo.
	Aprobado	Un requisito pasa a este estado cuando se ha aprobado por parte de todos los interesados, en este momento pasa a formar parte de una línea base.
	Rechazado	Un requisito es rechazado cuando no cumple al menos uno de los criterios para validar los requisitos.
Prioridad	Indica la prioridad del requisito.	Alta Tienen prioridad alta los requisitos que apoyan procesos priorizados como de prioridad alta, resuelven necesidades de los interesados de prioridad alta o son críticos para la arquitectura.
	Media	Tienen prioridad media, los requisitos que apoyan procesos

---

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

				priorizados como de prioridad media y resuelven necesidades de los interesados de prioridad media.
			Baja	Son los requisitos cuya implementación constituye un valor añadido al alcance previsto.
Breve descripción		Describe brevemente al requisito en términos de lo que debe hacer el sistema.	N/A	N/A
Reusabilidad	Origen	<p>Cuando existe una Especificación de Requisitos de Software que se está usando como entrada al modelado se completa este atributo.</p> <p>El origen es el nombre del sistema del que proviene el modelo, por ejemplo: UCID, VERSAT.</p>	N/A	N/A
	%	A partir de la evaluación del	0-100%	N/A

	especialista funcional se establece el % de reusabilidad del requisito.		
Versión objetivo	Establece en qué versión del sistema se debe implementar el requisito.	N/A	N/A
Referencia	Indica la dirección en el repositorio de control de versiones del documento en el que está descrito el requisito.	N/A	N/A

### 2.5.3.6 Componente

Tabla 11. Componente.

Atributo	Descripción	Posibles valores	Descripción
Id	Identifica de manera única al elemento de seguimiento.	N/A	N/A
Estado	Establece en qué estado se encuentra el componente.	Identificado	Cuando se define un componente este se registra, con una breve descripción, en el estado identificado.
		Diseñado	Un componente pasa a este estado cuando se

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

			ha completado el diseño del mismo en la plantilla de diagrama de clases.
		Implementado	Un requisito pasa a este estado cuando se ha concluido su implementación.
Prioridad	Indica la prioridad del componente.	Alta	Tienen prioridad alta los componentes que implementan requisitos con prioridad alta o son críticos para la arquitectura del sistema.
		Media	Tienen prioridad media los componentes que implementan requisitos con prioridad media.
		Baja	Son los componentes cuya implementación constituye un valor añadido al alcance previsto.
Breve descripción	Describe brevemente al componente en términos de su responsabilidad	N/A	N/A

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

		arquitectónica.		
Reusabilidad	Origen	Cuando existe un componente con funcionalidades similares se debe evaluar su reusabilidad. Se indica el nombre del componente.	N/A	N/A
	%	A partir de la evaluación de los arquitectos de sistema se establece el % de reusabilidad del componente.	0-100%	N/A
Versión objetivo		Establece en qué versión del sistema se debe implementar el componente.	N/A	N/A
Referencia		Indica la dirección en el repositorio de control de versiones del documento en el que está descrito el componente.	N/A	N/A

2.5.3.7 Solicitud de cambio

Tabla 12. Solicitud de cambio.

Atributo	Descripción	Posibles valores	Descripción
Id	Identifica de manera única al elemento de seguimiento.	N/A	N/A
Estado	Establece en qué estado se encuentra la incidencia.	Abierta	Cuando se registra una solicitud de cambio su estado es: abierta.
		Asignada	Si la solicitud de cambio procede se asigna un responsable de resolverla y esta pasa a: asignada.
		Resuelta	El responsable de resolver la solicitud de cambio cuando termina de hacerlo la pasa a este estado.
		Aprobada	Cuando se ha comprobado que efectivamente se resolvió la no conformidad esta pasa a: aprobada.

## CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

		Rechazada	Si la solicitud de cambio no procede se registra y pasa a: rechazada.
Prioridad	Indica la prioridad de la incidencia.	Alta	Hasta que se resuelva la solicitud de cambio el sistema está fuera de funcionamiento.
		Media	Hasta que se resuelva la no conformidad requisitos calificados como de prioridad alta o media no se podrá utilizar.
		Baja	Solicitudes de cambio que no afectan el funcionamiento del sistema, ni de requisitos calificados como de prioridad alta o media.
Breve descripción	Describe brevemente al proceso de negocio en términos de sus objetivos y entradas y salidas principales.	N/A	N/A
Referencia	Indica la dirección en el repositorio de control de versiones del	N/A	N/A

---

documento en el que  
está descrita la  
solicitud de cambio.

---

#### 2.5.4 PASO 4. INSTALAR Y CONFIGURAR LA HERRAMIENTA DE GESTIÓN DE REQUISITOS

Conocer la trazabilidad de un requisito puede llegar a ser una tarea compleja y extensa si se realiza manualmente. A partir de este punto en el procedimiento para el seguimiento a los requisitos se propone hacer uso de una herramienta de gestión de requisitos para agilizar y optimizar el procedimiento propuesto, en este caso para el CEIGE se propone hacer uso de EA en su versión 7.1.

#### 2.5.5 PASO 5. REGISTRAR ELEMENTOS DE SEGUIMIENTO A LOS REQUISITOS

Este paso consiste en registrar un elemento de seguimiento; o sea, llevar a la herramienta de gestión de requisitos propuesta todos los elementos descritos.

Al concluir la descripción de un elemento de seguimiento se registra en la herramienta utilizada, en este caso EA y pasa a formar parte del proceso de seguimiento a los requisitos.

Cuando se realiza esta tarea se especifican los atributos de dicho elemento, por ejemplo, la prioridad que puede ser baja, media o alta.

**Tabla 13. Paso 5. Registrar elementos de seguimiento.**

<b>Paso 5</b>	
<b>Nombre</b>	Registrar elementos de seguimiento a los requisitos.
<b>Objetivos</b>	Registrar un elemento de seguimiento en la herramienta.
<b>Roles</b>	Analista.
<b>Otros participantes</b>	N/A



<b>Artefactos</b>	N/A
-------------------	-----

2.5.6 PASO 6. DEFINIR LAS DEPENDENCIAS ENTRE LOS ELEMENTOS DE SEGUIMIENTO

Cuando es registrado un elemento de seguimiento, es necesario establecer las dependencias entre este y los elementos registrados anteriormente.

Esta tarea puede ser realizada inmediatamente después de ser registrado el elemento de seguimiento por el analista encargado de realizar esta actividad. El nuevo elemento insertado es relacionado con los registrados con anterioridad mediante una matriz de relaciones.

Gráficamente esta es una matriz donde los ejes son los elementos de seguimiento y las intercepciones, en dependencia si están marcadas o no, indican relación entre los dos elementos. En la siguiente figura se muestra un ejemplo de matriz de relaciones.

	Activar objetivo	Adicionar objetivo	Buscar objetivo	Desactivar objetivo	Eliminar objetivo	Exportar objetivos	Importar objetivos	Imprimir objetivos	Listar objetivos	Modificar objetivo	Realizar búsqueda avanzada de	Ver historial del objetivo
Aprobación	X		X			X			X			
Ejecutar sistema de control		X			X				X			
Establecer FIP				X			X				X	
Establecer plan		X				X			X			
Establecer sistema de control	X			X							X	
Puntualizar plan			X				X	X		X		X

**Figura 3. Matriz de relaciones.**

La siguiente tabla es un resumen para este paso.

**Tabla 14. Paso 6. Definir las dependencias entre los elementos de seguimiento.**

<b>Paso 6</b>	
<b>Nombre</b>	Definir las dependencias entre los elementos de seguimiento.
<b>Objetivos</b>	Establecer relaciones entre los elementos que son parte del proceso de seguimiento a los requisitos.
<b>Roles</b>	Analista.
<b>Otros responsables</b>	N/A
<b>Artefactos</b>	Matriz de relaciones.

#### 2.5.7 PASO 7. CREAR LA LÍNEA BASE DE LOS ELEMENTOS DE SEGUIMIENTO

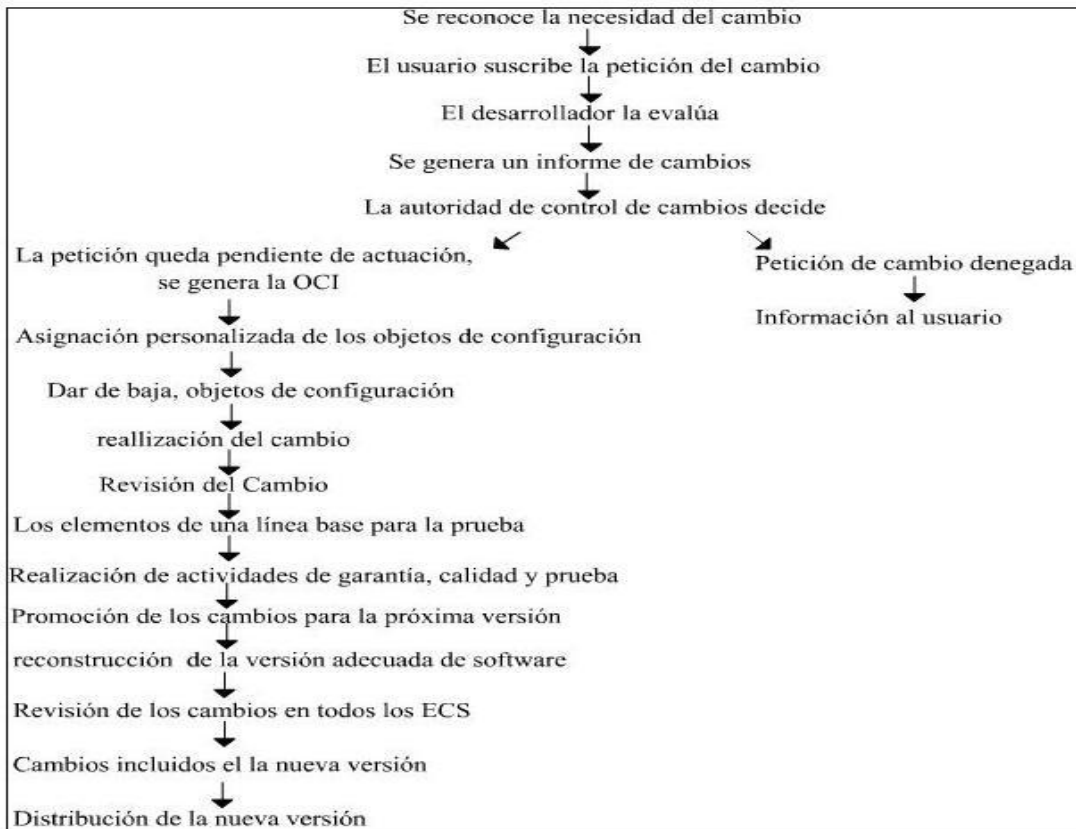
Pressman ha definido una línea base como un concepto de gestión de configuración de software que nos ayuda a controlar los cambios sin impedir los cambios justificados (Pressman, 2002).

La IEEE (Estándar IEEE 610.12-1990) define una línea base como: *“Una especificación o producto que se ha revisado formalmente y sobre los que se ha llegado a un acuerdo, y que de ahí en adelante sirve como base para un desarrollo posterior y que puede cambiarse solamente a través de procedimientos formales de control de cambios.”*

Conociendo los conceptos dados por diferentes autores se puede resumir la definición de línea base como un producto que ha sido revisado formalmente y sólo puede ser cambiado a través de un procedimiento de control de cambios.

En el contexto de la ingeniería del software se ha definido una línea base como un punto de referencia en el desarrollo del software y que queda marcado por el envío a este de uno o más elementos de configuración de software (ECS) y la aprobación de ECS obtenido mediante una revisión técnica formal. (Pressman, 2002)

Para cada cambio aprobado se genera una orden de cambio de ingeniería (OCI). La OCI describe el cambio a realizar, las restricciones que se deben respetar y los criterios de revisión y de auditoría. Además, debe hacerse una evaluación de cada uno, así como un seguimiento y revisión de los mismos. El siguiente diagrama muestra más explícitamente el proceso de cambio (Pressman, 2002).



**Figura 4. Proceso de control de cambio según Pressman.**

Siguiendo el concepto proporcionado por Pressman sobre línea base y control de cambio se propone seguir el siguiente criterio sobre estos temas en el procedimiento propuesto en el presente trabajo de diploma.

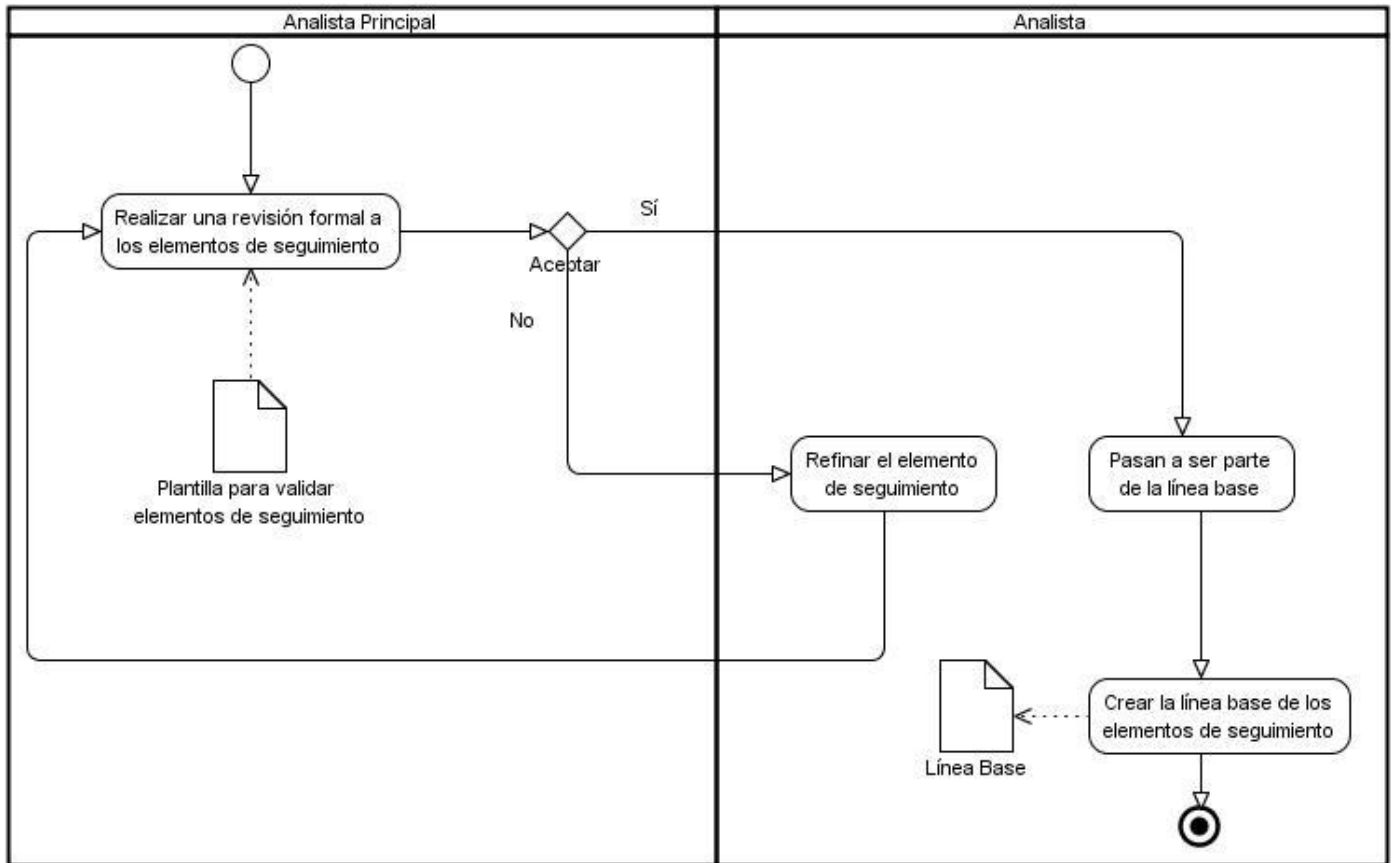
Después de haber identificado los elementos de seguimiento, especificados los atributos propios de cada cual, y haber definido las relaciones entre estos y otros elementos pasan a la línea base después de

realizar una revisión formal a la descripción de los elementos de seguimiento y ser aceptados por el analista principal. Si no es aceptado algún elemento, es necesario refinarlo y hacer una nueva revisión.

Que un elemento de seguimiento sea aceptado o rechazado depende de los términos definidos para declararlos válidos. Para un requisito, por ejemplo, estos criterios de validación están definidos en la Plantilla para validar elementos de seguimiento (Minet, 2009), esta plantilla es generada para dictar un criterio sobre el estado del requisito, o sea si este es aprobado o rechazado.

Criterios de validación para requisitos:

- ¿El proveedor del requisito es un proveedor válido?
- ¿El requisito está identificado como único?
- ¿El requisito es modificable?
- ¿El requisito no es ambiguo?
- ¿El requisito está completo?
- ¿El requisito es congruente con otros REQ relacionados?
- ¿El requisito puede ser implementado?
- ¿El requisito puede ser probado?
- ¿El resultado de la evaluación de impacto es positivo?
- ¿El requisito está correcto?
- ¿El requisito es traceable?



**Figura 5. Flujo de actividades Crear línea base.**

Para realizar un cambio sobre los elementos que forman parte de la línea base se debe seguir el proceso de control de cambio descrito a continuación. Primero, el analista debe hacer una solicitud de cambio formal a través del Formulario de solicitud de cambio, ver Anexo 1, especificando el problema por el cual se debe realizar y la descripción del cambio propuesto. Posteriormente, debe esperar una orden de cambio, o sea, una aprobación por parte del comité de control de cambios, el cual debe hacer con anterioridad una valoración sobre el impacto del cambio en el sistema y decidir si es factible o no realizarlo.

Criterios para declarar factible un cambio:

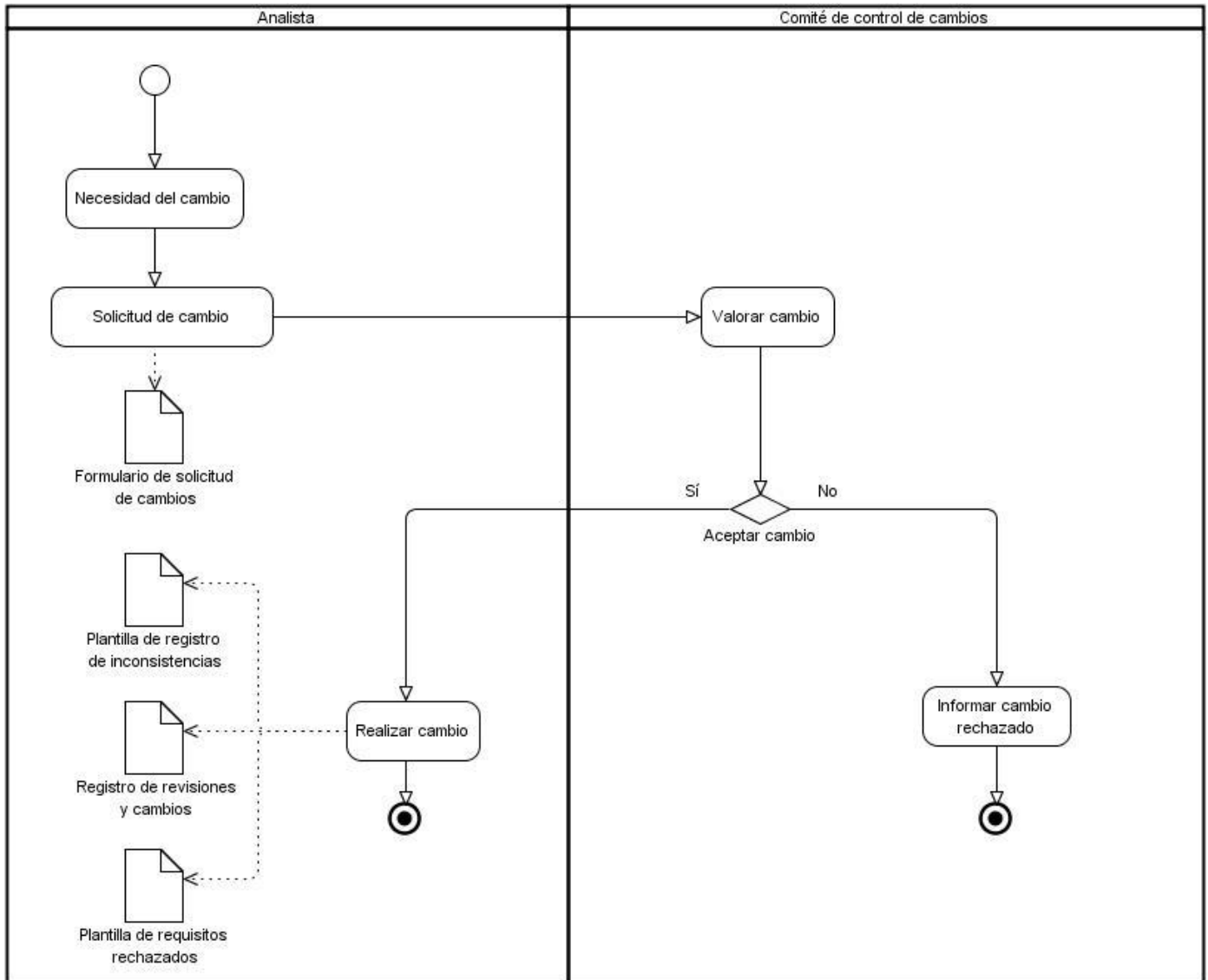
- Estado del cronograma.

- Utilidad de la funcionalidad.
- Prioridad de la funcionalidad.
- Complejidad de la solución.

Si el cambio no es aceptado se informa al analista solicitante el por qué es rechazado. Si es aceptado se deben generar los siguientes artefactos para proceder a realizar el cambio:

- Plantilla de requisitos rechazados (Anexo 4): describir por qué se ha rechazado el requisito.
- Registro de revisiones y cambios (Anexo 5): registrar la fecha de solicitud, la fecha de solución, el responsable del cambio.
- Plantilla de registro de inconsistencias (Anexo 6): describir las inconsistencias encontradas.

El proceso es descrito en detalle en la siguiente figura.



**Figura 6. Flujo de actividades Control de cambio.**

En la siguiente tabla se realiza un resumen del proceso descrito anteriormente.

**Tabla 15. Paso 7. Crear línea base de los elementos de seguimiento.**

**Paso 7**

<b>Nombre</b>	Crear línea base de los elementos de seguimiento.
<b>Objetivos</b>	<ul style="list-style-type: none"> <li>• Crear la línea base.</li> <li>• Establecer un proceso de control de cambio.</li> </ul>
<b>Roles</b>	Analista.
<b>Otros participantes</b>	Comité de control de cambios.
<b>Artefactos</b>	Línea base.

#### 2.5.8 PASO 8. OBTENER INFORMACIÓN DE SEGUIMIENTO

Parte fundamental en el proceso de seguimiento a los requisitos es la obtención de información de los elementos de seguimiento. En el procedimiento propuesto esta información se resume principalmente en la matriz de relaciones, el estado de los elementos y la línea base.

**Tabla 16. Paso 8. Obtener información de seguimiento.**

<b>Paso 8</b>	
<b>Nombre</b>	Obtener información de seguimiento.
<b>Objetivos</b>	Obtener información en cuanto al estado de: <ul style="list-style-type: none"> <li>• Matriz de relaciones</li> <li>• Línea base</li> <li>• Elementos de seguimiento</li> </ul>
<b>Roles</b>	Analista.
<b>Otros participantes</b>	N/A



<b>Artefactos</b>	N/A
-------------------	-----

Obtener información de la matriz de relaciones consiste en conocer la trazabilidad de los elementos de seguimiento. En cuanto a la línea base esta información consiste en conocer si se han realizado cambios en el modelo desde que esta fue creada. Para los elementos de seguimiento consiste en conocer el estado que estos se encuentran. Esta información puede ser consultada directamente en la herramienta utilizada o pueden generarse reportes que la contengan.

La información de seguimiento es utilizada para evaluar el impacto en el sistema que producen los cambios en los elementos de seguimiento. Ayuda a estimar con mejor precisión el tiempo que resta en el desarrollo del proyecto. También ofrece una visión de la complejidad del sistema y permite priorizar tareas en el desarrollo del proyecto como puede ser generar artefactos. Además, es una base de conocimiento que puede ser tomada como experiencia para futuros proyectos.

## **2.6 CONCLUSIONES PARCIALES**

En el presente capítulo se describió en detalle el procedimiento propuesto para mantener el seguimiento a los requisitos en el CEIGE. Fueron definidos ocho pasos para realizar este proceso, especificando los artefactos generados y los roles que intervienen en este.

## Capítulo 3. VALIDACIÓN DEL PROCEDIMIENTO

### 3.1 INTRODUCCIÓN

La aplicación del procedimiento propuesto se realizará en uno de los proyectos del CEIGE y se validará mediante encuestas a los integrantes del proyecto en cuestión. Se ejecutarán cada uno de los pasos descritos en el procedimiento, hasta abarcar la creación de líneas base, el proceso de control de cambios y la obtención de información de seguimiento. Finalmente serán estudiados los resultados arrojados durante la aplicación del procedimiento para conocer la aceptación del mismo.

### 3.2 APLICACIÓN DEL PROCEDIMIENTO

Una vez concluido el proceso de elaboración del procedimiento propuesto en el Capítulo 2 del presente trabajo de diploma se procede a la validación del mismo. Para dar cumplimiento a dicha actividad se ha decidido desarrollar la aplicación de dicho procedimiento. Este será aplicado en el proyecto Dirección por Objetivos perteneciente a la línea Planificación del CEIGE y se hará uso de Enterprise Architect como herramienta de gestión de requisitos para darle cumplimiento a esta tarea.

En este capítulo se aborda lo referente al análisis de los resultados, donde se hace un resumen de los principales aportes y ventajas que proporciona la aplicación del procedimiento antes definido haciendo uso de la herramienta seleccionada en la Fundamentación Teórica del presente trabajo de diploma.

Para realizar la aplicación del procedimiento se tuvo la participación de un analista principal teniendo en cuenta la experiencia que este presenta, además de dos analistas con habilidades en EA y conocimientos sobre Ingeniería de software.

Las primeras tres actividades definidas para el procedimiento; o sea, establecer los elementos de seguimiento, sus relaciones y atributos, se realizan al comienzo del proceso de seguimiento a los requisitos y no es necesario realizarlas nuevamente. El resultado es guardado en la Guía para el seguimiento a los requisitos para ser consultado posteriormente.

En la tabla que se muestra a continuación están representados los elementos y las cantidades en que fueron llevados a la herramienta. Una vez insertados se establecieron las dependencias entre estos, realizando de esta forma los pasos 5 y 6 del procedimiento.

**Tabla 17. Elementos de seguimiento seleccionados.**

<b>Elementos de seguimiento</b>	<b>Cantidad</b>
<b>Procesos de negocio</b>	6
<b>Requisitos funcionales</b>	236
<b>Requisitos no funcionales</b>	3
<b>Agrupaciones de requisitos</b>	12
<b>Componentes</b>	16

Una vez insertados los elementos de seguimiento se realizó una revisión formal a los mismos para verificar si cumplían con las condiciones necesarias para formar parte de la línea base. Para esto se siguieron los criterios de validación definidos para cada elemento de seguimiento, para los procesos de negocio, por ejemplo, se siguieron criterios como:

- ¿El proveedor del proceso es válido?
- ¿El proceso está unívocamente identificado?
- ¿El resultado de la evaluación de impacto es positivo?

Los elementos resultaron como aceptados y pasaron a formar parte de la línea base.

Posteriormente, se decidió realizar cambios a dos de dichos elementos: al componente Gestionar actividades y a la agrupación de requisitos Gestionar grupos de usuarios. Para realizar este proceso se debió generar un Formulario de solicitud de cambio para cada elemento, ver Anexo 1.

El cambio propuesto al componente consistía en adicionar la funcionalidad Agregar repeticiones anuales. El comité de control de cambios, después de realizar un estudio del impacto que este causaría, decidió que no procedía cambiarlo debido a la baja prioridad de la nueva funcionalidad y al estado apremiante del cronograma.

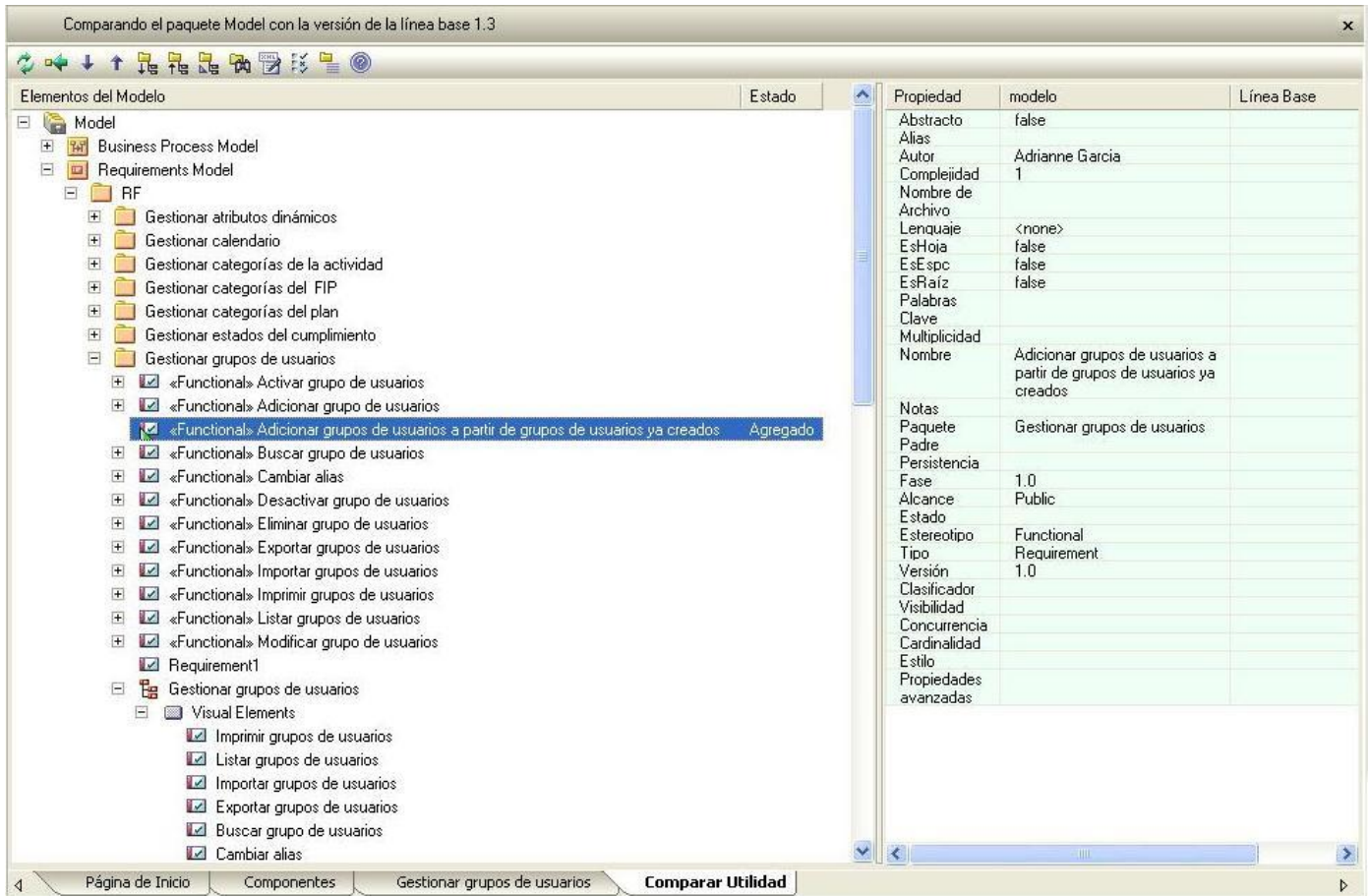
El cambio a la agrupación de requisitos consistía en adicionar la funcionalidad Agregar grupos de usuarios a partir de grupos de usuarios ya creados. En este caso el comité de control de cambios decidió realizar el cambio debido a la alta utilidad de la funcionalidad y a la baja complejidad de la solución. Para esto se generaron los siguientes artefactos:

- Plantilla de requisitos rechazados.
- Registro de revisiones y cambio.
- Plantilla de registro de inconsistencias.

En cuanto al último paso en el procedimiento propuesto, o sea la obtención de información sobre el estado de los elementos de seguimiento, la línea base y la matriz de relaciones se procedió a utilizar la herramienta de gestión de requisitos.

Para obtener información sobre el estado de los elementos se procedió a generar reportes de tres elementos distintos; al proceso de negocio Aprobación, a la agrupación de requisitos Gestionar objetivos y al componente Gestionar grupos de usuarios lográndose la documentación deseada. En el Anexo 2 se muestra el reporte referente al proceso de negocio Aprobación; en este se encuentra, además, una tabla con todos los elementos con los que este proceso está relacionado.

Con los elementos de seguimiento; procesos de negocio, agrupaciones de requisitos, requisitos funcionales, requisitos no funcionales y componentes; existentes en el sistema se procedió a generar la línea base. A propósito de los cambios hechos a la agrupación de requisitos Gestionar grupos de usuarios se hizo uso de la opción de comparación, los resultados se muestran en la figura 25.



**Figura 7. Utilidad de comparación. Resultados.**

Para hacer uso práctico de la Matriz de relaciones se generó, primeramente, una matriz para crear dependencias entre los elementos que se encuentran en los paquetes Procesos de negocio y Gestionar objetivos, la matriz se muestra en la figura 26; seguidamente, para hacer más confiables los resultados se creó otra matriz para relacionar los paquetes Componentes y Gestionar grupos de usuarios, como se muestra en la figura 27.

### CAPÍTULO 3. VALIDACIÓN DEL PROCEDIMIENTO

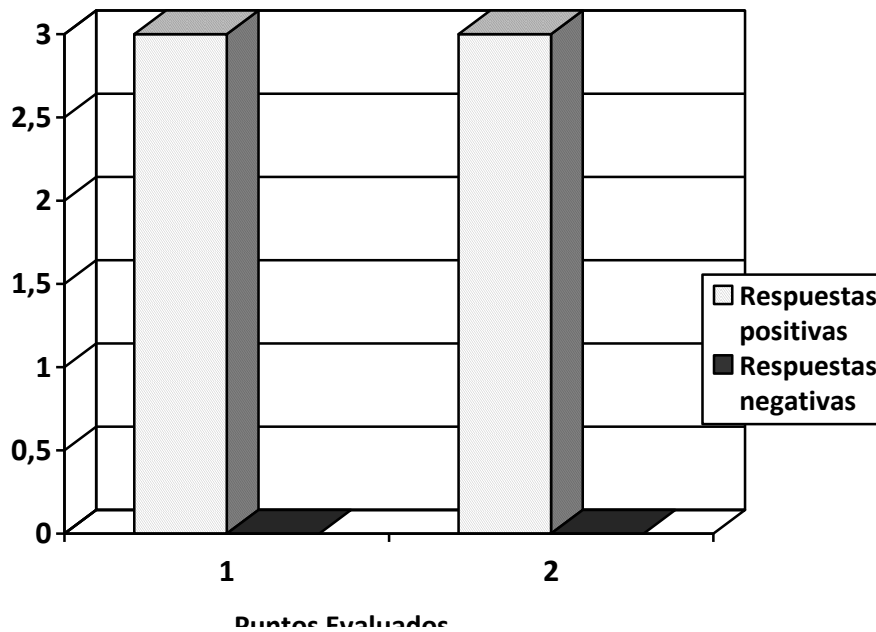
Origen: Business Process		Tipo: <Todos>		Tipo de: Asociación								
Destino: Gestionar objetivos		Tipo: <Todos>		Dirección: Origen -> Destino								
	Gestionar objetivos::Activar obje	Gestionar objetivos::Adicionar c	Gestionar objetivos::Buscar obj	Gestionar objetivos::Desactivar	Gestionar objetivos::Eliminar ob	Gestionar objetivos::Exportar oi	Gestionar objetivos::Importar ot	Gestionar objetivos::Imprimir ot	Gestionar objetivos::Listar objet	Gestionar objetivos::Modificar o	Gestionar objetivos::Realizar bu	Gestionar objetivos::Ver historie
Business Process Model::Apr...	X		X			X			X			
Business Process Model::Eje...		X			X				X			
Business Process Model::Est...				X			X				X	
Business Process Model::Est...		X				X			X			
Business Process Model::Est...	X			X							X	
Business Process Model::Pun...			X				X	X		X		X

**Figura 8. Matriz de relaciones Procesos de negocio y Gestionar objetivos.**



Utilidad de la propuesta

Para valorar este aspecto se tuvieron en cuenta las respuestas de los encuestados en los puntos 1 y 2, referidos a la importancia de desarrollar el seguimiento a los requisitos para el éxito de un proyecto de software y a la necesidad de definir un procedimiento para mantener el seguimiento a los requisitos acorde a las características de cada proyecto, respectivamente. Las respuestas a los puntos podían ser positivas o negativas y los resultados se muestran a continuación.

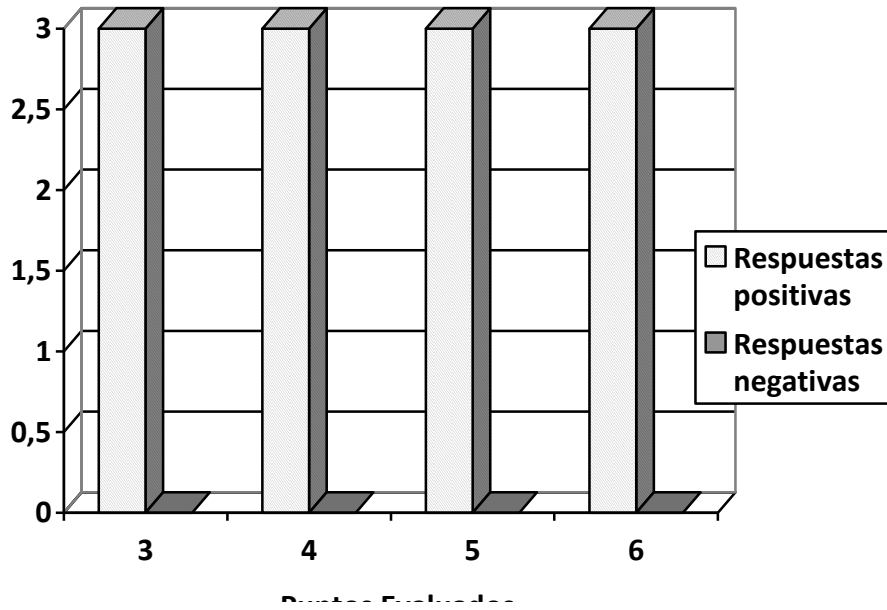


**Figura 10. Utilidad de la propuesta.**

Corrección del procedimiento

Para obtener una valoración al respecto se plantearon los puntos 3, 4, 5 y 6 a evaluar referidos a la corrección de las actividades, roles, artefactos y relaciones entre estos. Las respuestas a los puntos podían ser positivas o negativas y el resultado se muestra en el siguiente gráfico.





**Figura 11. Corrección del procedimiento.**

Las respuestas a los encuestados fueron positivas en cuanto a la utilidad de la propuesta y a la corrección del procedimiento propuesto. Además, la aplicación del punto 7 permitió conocer elementos de interés que posibiliten la mejora del Procedimiento para mantener el seguimiento a los requisitos en el CEIGE; estos consistían, principalmente, en no omitir algún paso propuesto y realizarlos en el orden establecido.

### **3.4 CONCLUSIONES PARCIALES**

Después de la puesta en práctica del procedimiento propuesto en el capítulo 2 y analizados los resultados se concluye que, en general, el procedimiento es calificado de válido, completo y efectivo para desarrollar el seguimiento a los requisitos en el CEIGE.

Al aplicar el procedimiento se hizo necesario registrar los elementos de seguimiento identificados en el proyecto Dirección por objetivos para generar toda la documentación requerida lográndose este objetivo.

## CONCLUSIONES

Realizando el marco teórico de la investigación se llegó a la conclusión que la Ingeniería de requisitos, específicamente, el Seguimiento a los requisitos es fundamental en el proceso de desarrollo de software por la importancia que aporta conocer la trazabilidad de los requisitos.

La necesidad de realizar el seguimiento a los requisitos de software en el CEIGE de forma óptima propició el desarrollo de un procedimiento que permita realizar esta actividad, siendo preciso definir roles, artefactos y actividades y organizarlos de manera que se obtuviera un proceso completo con el que se garantizara la trazabilidad de los requisitos de software.

Realizando la aplicación del procedimiento en el proyecto Dirección por Objetivos de la línea de Planificación del CEIGE y realizando su validación del mismo mediante encuestas a los participantes en esta se demostró que es posible obtener la trazabilidad de los elementos de seguimiento.

## RECOMENDACIONES

### **RECOMENDACIONES**

Se recomienda que el procedimiento propuesto sea utilizado por los proyectos del CEIGE.

## BIBLIOGRAFÍA

**Colectivo de autores. 2009.** Modelo de Desarrollo y Gestión de Proyectos de Software. s.l. : Unidad de Compatibilización Integración y Desarrollo De Software para la Defensa, 2009.

**CEIGE. 2009.** Guía para el seguimiento a los requisitos. Consultoría y Proyecto. 2009. 2.0.

**Chaves, Michael. 1997.** La Ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. 1997.

**Española, Rael Academia. 2010.** RAE. [En línea] 2010. <http://www.rae.es/rae.html>.

**IEEE. 1998.** Guide to Software Requirements Specifications. 1998.

**Jacobson, Ivar. 2000.** El Proceso Unificado de Desarrollo de Software. s.l. : Pearson Educación S.A, 2000.

**Normalización, Oficina Nacional de. 2006.** Directivas para la Aplicación de la NC ISO 9001:2001 al Software de Computación. Cuban National Bureau of Standards. 2006.

**Pinheiro, Francisco. 2000.** Formal and Informal Aspects of Requirements Tracing. 2000.

**Pressman, Roger. 2002.** Ingeniería del Software. Un enfoque práctico. s.l. : 5ª Edición, 2002, 2002.

**Reifer, Donald. 1994.** Encyclopedia of Software Engineering. 1994.

**Sommerville, Ian. 2002.** Ingeniería de Software. s.l. : Adison Wesley, 2002.

**Thayer, R. H. 1997.** Software Requirements Engineering. 1997.

**Agusti, Yadira Pérez.** [http://scielo.sld.cu/scielo.php?pid=S1024-94352008000300009&script=sci\\_arttext](http://scielo.sld.cu/scielo.php?pid=S1024-94352008000300009&script=sci_arttext).

**Broocs, Frederick P. 1987.** *No Silver Bullet: Essence and Accidents of Software Engineering.* 1987. Vol. 20.

**Miniet, Ing. Yanet Vega. 2009.** Plantilla para validar elementos de seguimiento a los requisitos. *Criterios*

## BIBLIOGRAFÍA

*para validar elementos de seguimiento a los requisitos.* s.l. : CEIGE, 2009

**Hidalgo, Joan Gabriel Rodríguez. 2009.** *Herramienta para la obtención de requisitos en el desarrollo de software a distancia.* UCI. 2009. Tesis de pregrado.

**Pescador, Elena Sánchez. 2009.** *Análisis e implementación de una herramienta de Gestión de requisitos para la Gestión de servicios.* Universidad Carlos III. Madrid : s.n., 2009.

**Vázquez, Ameirys Betancourt. 2007.** *Modelación del Negocio, Requisitos y Análisis para el Sistema del Servicio Autónomo de los Registros y Notarías en la República Bolivariana de Venezuela.* UCI. 2007. Tesis de pregrado.

## **ANEXO 1. FORMULARIO DE SOLICITUD DE CAMBIO**

### **1 Introducción**

#### **Objetivo**

El objetivo de este documento registrar los cambios solicitados al producto.

#### **Alcance**

<<Versión del producto a la que solicita realizar el cambio>>

#### **Definiciones y acrónimos**

N/A.

#### **Referencias**

N/A.

### **2 Identificación**

#### **Número:**

[Número consecutivo identificador de la solicitud de cambio]

#### **Tipo:**

[Tipo de Solicitud de Cambio: error o mejora]

#### **Título:**

[Título de la Solicitud de Cambio que refleje de manera sintetizada el contenido de la solicitud]

#### **Fecha de creación:**

[Fecha de creación de la Solicitud de Cambio]

**Creada por:**

[Nombre de la persona que creó la solicitud de cambio, y contacto de la misma]

**Prioridad:**

[Prioridad de la solicitud de cambio, esta puede ser alta, media o baja en dependencia de la importancia del mismo]

### **3 Problema actual**

**Descripción:**

[Breve descripción de la solicitud de cambio que abarque los aspectos más importantes del mismo]

**Como repetir:**

[Secuencia de pasos que permitan repetir el problema]

**Nuevos requisitos:**

[Listado de nuevos requisitos que desean ser agregados]

**Condiciones bajo las que fue observado el problema:**

[Condiciones de software bajo las que fue detectado el problema]

**Ambiente actual: hardware:**

[Hardware correspondiente al ordenador donde se detectó el problema]

**Sistema operativo:**

[Sistema operativo del ordenador donde se detectó el problema]

### **4 Cambio propuesto**

**Cambio propuesto por el creador**



**Descripción:**

[Descripción del cambio propuesto por parte del creador de la solicitud de cambio]

**Costo estimado:**

[Costo estimado del cambio por parte del creador de la solicitud]

**Cambio propuesto por el equipo de desarrollo**

**Acción:**

[Explicación de la decisión tomada respecto al cambio. Pudiera ser aplazarlo para una próxima iteración del software]

**Decisión tomada:**

[Aceptar, rechazar]

**Elementos de configuración afectados:**

[Elementos de configuración que se ven afectados por el pedido de cambio, cuántos, cuáles y cómo son afectados (Apoyarse en la Matriz de Trazabilidad)]

**Errores corregidos:**

[Errores corregidos durante la implementación de la solicitud]

**Nuevas funcionalidades:**

[Nuevas funcionalidades que pudieron ser agregadas durante la implementación de la solicitud]

## **5 Resolución**

**Tiempo y costo estimado del cambio propuesto:**

[Tiempo y costo estimado por parte del equipo de desarrollo para la realización del cambio]

**Desarrollador(es):**

[Nombre(s) de desarrollador(es) de la solicitud de cambio]

## **ANEXO 2. REPORTE APROBACIÓN:**

### Documentación del Modelo

#### **Detalle del Modelo**

Este documento provee una completa perspectiva general de los detalles de todos los elementos. Para reportes más simples y enfocados, simplemente copie esta plantilla inicial y deseccione las secciones no requeridas.

#### **Aprobación**

*Tipo:* **Actividad**

*Estado:* Proposed. *Versión* 1.0. *Fase* 1.0.

*Paquete:* Business Process Model *Palabras claves:*

*Detalle:* Creado el 08/05/2010. Última modificación el 08/05/2010.

*GUID:* {0DBCCFB7-0E7F-43f7-80A6-A73BFAEB0F43}

Aprobar un documento.

#### **Propiedades Personalizadas**

- isReadOnly = Falso
- precondition =
- isSingleExecution = Falso

## ANEXOS

### Propiedades Personalizadas

- postcondition =
  
- parameterName =

### Conexiones

<b>Conector</b>	<b>Origen</b>	<b>Destino</b>	<b>Notas</b>
<b><u>Asociación</u></b> Sin especificar	Public Aprobación	Public Listar objetivos	
<b><u>Asociación</u></b> Sin especificar	Public Aprobación	Public Exportar objetivos	
<b><u>Asociación</u></b> Sin especificar	Public Aprobación	Public Buscar objetivo	
<b><u>Dependencia</u></b> Origen -> Destino	Public Aprobación	Public Activar atributos dinámicos	
<b><u>Asociación</u></b> Sin especificar	Public Aprobación	Public Activar objetivo	

Conector	Origen	Destino	Notas
<u>Asociación</u> Sin especificar	Public Aprobación	Public Adicionar atributo dinámico	

### ANEXO 3. ENCUESTA A LOS PARTICIPANTES EN LA APLICACIÓN DEL PROCEDIMIENTO.

**Evaluación del Procedimiento para mantener el seguimiento a los requisitos de software.**

1. ¿Considera que es importante para el éxito de un proyecto de software mantener el seguimiento a los requisitos?

Si: \_\_\_\_ No: \_\_\_\_ ¿Por qué?: \_\_\_\_\_

2. ¿Considera que es necesario definir un procedimiento para mantener el seguimiento a los requisitos que se adecúe a las características del proyecto?

Si: \_\_\_\_ No: \_\_\_\_ ¿Por qué?: \_\_\_\_\_

**– Evalúe el procedimiento propuesto**

3. ¿Son adecuadas las actividades definidas en el procedimiento?

Si: \_\_\_\_ No: \_\_\_\_ ¿Por qué?: \_\_\_\_\_

4. ¿Son adecuados los artefactos propuestos en el procedimiento?

Si: \_\_\_\_ No: \_\_\_\_ ¿Por qué?: \_\_\_\_\_

5. ¿Los roles están en correspondencia con la labor que desempeñan?

Si: \_\_\_\_ No: \_\_\_\_ ¿Por qué?: \_\_\_\_\_

6. ¿Hay una adecuada relación entre roles, actividades y artefactos en el procedimiento?

Si: \_\_\_\_ No: \_\_\_\_ ¿Por qué?: \_\_\_\_\_

7. Elabore un comentario general sobre el procedimiento que está siendo evaluado que aporte elementos a la mejora del mismo.

## **ANEXO 4. PLANTILLA DE REQUISITOS RECHAZADOS**

### **Introducción**

#### **Objetivo**

El objetivo de este documento es describir porqué se ha rechazado el requisito <<Nombre del requisito>> y establecer la recomendaciones necesarias para su corrección y aprobación.

#### **Alcance**

Proyecto <<Nombre del proyecto>>.

#### **Definiciones y acrónimos**

N/A.

#### **Referencias**

N/A.

## Reporte de trazabilidad

### Datos generales

Solicitado por	Fecha de solicitud	Rechazado por
<<Nombre y apellidos>> <<Cargo>>	<<dd/mm/aaaa>>	<<Nombre y apellidos>> <<Cargo>>

### Criterio de fallo

[Debe de seleccionarse de los siguientes motivos uno:

Proveedor de requisito no válido.

El requisito no es único.

El requisito no es modificable.

El requisito es ambiguo.

El requisito está incompleto.

Incongruencia con otros requisitos.

El requisito no puede ser implementado.

El requisito no puede ser probado.

El requisito está incorrecto.

El requisito no es traceable.

La evaluación de impacto proporcionó resultados negativos.]

<b>Causa</b>	[Debe describirse la causa por la que no cumple el criterio de aceptación.]
<b>Propuesta</b>	[Debe proponerse una manera diferente de enfocar el requisito. De lo contrario, su esclarecimiento, eliminación, posposición u otra acción a considerarse.]

### Criterio de fallo

[Debe de seleccionarse de los siguientes motivos uno que no coincida con el anterior:

Proveedor de requisito no válido.

El requisito no es único.

El requisito no es modificable.

El requisito es ambiguo.

El requisito está incompleto.

Incongruencia con otros requisitos.



**ANEXO 6. PLANTILLA DE REGISTRO DE INCONSISTENCIAS**

Nº	Fuente de la Inconsistencia	Descripción de la Inconsistencia	Fecha	Lista de elementos inconsistentes	Estado	Responsable
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						