

Universidad de las Ciencias Informáticas



Facultad 15

Título: “Levantamiento de requisitos de los Módulos Personal de apoyo y Reportes y búsquedas del subsistema Gestión de Cuadros y Personal de Apoyo del proyecto Sistema de Gestión Fiscal”

Trabajo de Diploma para optar por el título de Ingeniero Informático

Autor: Inelsis Valdespino Paez

Tutor: Ing. Maylín Bacallao Martínez

Asesores:

- ✓ Ing. Dailín Benavides Jorge
- ✓ Ing. Yelena Hernández Estrada

Curso 2009-2010

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Inelsis Valdespino Paez

Ing. Maylín Bacallao Martínez

AGRADECIMIENTOS

A nuestro Comandante en Jefe Fidel Castro, por ser el motor impulsor en la realización de un sueño: la educación en Cuba.

A mi querida y estimada amiga de batallas, Eliska, sin su ayuda y apoyo tal vez este trabajo no hubiera alcanzado su propósito.

A Yudenia por orientarme cuando me sentí perdida y guiarme aún sin conocerme lo suficiente; su ayuda fue fundamental para mí.

A Yelena por la paciencia que me tuvo durante el desarrollo de este trabajo y por la ayuda que me brindó.

A todas las personas que de una forma u otra se preocuparon porque el estado del presente trabajo y por darme aliento. También quisiera agradecer a las que influyeron en mi formación universitaria. ¡Gracias a todos!

DEDICATORIA

A mi madre por brindarme siempre su apoyo incondicional, su confianza y por hacer de mí la persona que soy.

A mi novio por estar siempre conmigo en los buenos y malos momentos y por levantarme el ánimo cuando más lo necesité.

A mi padre y a mi hermana por la confianza que han depositado en mí durante estos largos años de estudio... ¡Los quiero mucho!

RESUMEN

En el presente trabajo se realiza un estudio de las Metodologías para el desarrollo de software, Lenguajes de Modelado, Herramientas CASE y Herramientas de modelado de prototipos de interfaz a utilizar, así como las etapas de la ingeniería de requisitos y técnicas que se aplican durante las mismas. Se propone además una guía para llevar a cabo el levantamiento de los requisitos de los módulos Personal de apoyo y Reportes y búsquedas del subsistema Gestión de Cuadros y Personal de Apoyo del proyecto Sistema de Gestión Fiscal. Para dar cumplimiento a la misma, se desarrollan artefactos de los flujos de trabajo Modelado del negocio y Requisitos, como por ejemplo: el Diagrama de casos de uso del negocio, Modelo de objeto, Diagrama de casos de uso del sistema y los Prototipos de interfaz de usuario. Por último se evalúa el trabajo realizado mediante la aplicación de métricas y técnicas de validación que garantizan la calidad de los artefactos generados, los cuales arrojaron resultados positivos.

Palabras claves: Personal de apoyo, Reportes y búsquedas, Ingeniería de Requisitos, Herramientas, Metodología, Métricas.

Índice

INTRODUCCIÓN	1
CAPÍTULO 1. “FUNDAMENTACIÓN TEÓRICA”	5
1.1 INTRODUCCIÓN	5
1.2 SISTEMAS RELACIONADOS CON LOS MÓDULOS PERSONAL DE APOYO Y REPORTES Y BÚSQUEDAS EN EL MUNDO	5
1.3 PROCESO DE DESARROLLO DE SOFTWARE	7
1.4 METODOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE	7
1.4.1 <i>Metodologías Ágiles</i>	7
1.4.1.1 eXtreme Programing (XP)	8
1.4.2 <i>Metodologías Tradicionales</i>	8
1.4.2.1 Microsoft Solution Framework (MSF)	9
1.4.2.2 RUP (Rational Unified Process)	9
1.5 HERRAMIENTAS CASE	11
1.5.1 <i>Rational Rose</i>	11
1.5.2 <i>Visual Paradigm 6.4 for UML Enterprise Edition</i>	12
1.6 HERRAMIENTAS DE MODELADO DE PROTOTIPOS DE INTERFAZ DE USUARIO	12
1.6.1 <i>Microsoft Office Visio 2007</i>	13
1.6.2 <i>Visual Paradigm for UML 6.4</i>	13
1.6.3 <i>Axure RP 5.5</i>	13
1.7 LENGUAJES PARA EL MODELADO	14
1.7.1 <i>IDEFO</i>	15
1.7.2 <i>BPMN</i>	15
1.7.3 <i>Lenguaje de Modelado Unificado (UML)</i>	16
1.8 INGENIERÍA DE REQUISITOS	17
1.8.1 <i>Etapas de desarrollo de la ingeniería de requisitos</i>	19
1.8.2 <i>Técnicas de la ingeniería de requisitos</i>	22
1.8.2.1 <i>Técnicas para la captura de requisitos</i>	22
1.8.2.2 <i>Técnicas de validación de requisitos</i>	25
1.9 CONCLUSIONES PARCIALES	26
CAPÍTULO 2. “SOLUCIÓN PROPUESTA”	27
2.1 INTRODUCCIÓN	27
2.2 PLANIFICACIÓN DEL LEVANTAMIENTO DE REQUISITOS	27
2.3 MODELADO DEL NEGOCIO DEL MÓDULO PERSONAL DE APOYO	31
2.3.1 <i>Reglas del negocio</i>	32
2.3.2 <i>Actores del negocio</i>	32
2.3.3 <i>Trabajadores del negocio</i>	33
2.3.4 <i>Casos de uso del negocio</i>	33
2.3.5 <i>Diagrama de casos de uso del negocio</i>	34
2.3.6 <i>Realización de casos de uso del negocio</i>	34
2.3.7 <i>Diagrama de Actividades</i>	39
2.3.8 <i>Modelo de objetos</i>	39
2.4 MÓDULO REPORTES Y BÚSQUEDAS	40
2.5 FLUJO DE TRABAJO DE REQUISITOS	40
2.5.1 <i>Requisitos funcionales</i>	40
2.5.1.1 <i>Requisitos funcionales definidos para el módulo Personal de apoyo</i>	40
2.5.1.2 <i>Algunos requisitos funcionales definidos para el módulo Reportes y búsquedas</i>	41

2.5.2 Requisitos no funcionales.....	41
2.5.3 Actores del sistema.....	42
2.5.4 Casos de uso del sistema.....	43
2.5.5 Diagrama de casos de uso del sistema	44
2.5.5.1 Diagrama de casos de uso del sistema definido para el módulo Personal de apoyo.....	44
2.5.5.2 Diagrama de casos de uso del sistema definido para el módulo Reportes y búsquedas	45
2.5.6 Realización de los casos de uso del sistema	45
2.5.6.1 Realización de los casos de uso del módulo Personal de apoyo.....	46
2.5.6.2 Realización de los casos de uso del módulo Reportes y búsquedas	49
2.6 PATRONES DE CASOS DE USO UTILIZADOS.....	50
2.7 CONCLUSIONES PARCIALES	52
CAPÍTULO 3. VALIDACIÓN	53
3.1 INTRODUCCIÓN	53
3.2 VALIDACIÓN MEDIANTE PROTOTIPOS NO FUNCIONALES	53
3.3 REVISIONES	53
3.4 VALIDACIÓN MEDIANTE LISTAS DE CHEQUEO	54
3.5 MÉTRICAS.....	54
3.5.1 Métricas de la calidad de la especificación	55
3.5.3 Métricas para el Diagrama de Casos de Uso.....	57
3.6 MATRIZ DE TRAZABILIDAD	62
3.7 ACTA DE LIBERACIÓN	63
3.8 CONCLUSIONES PARCIALES	63
CONCLUSIONES GENERALES.....	64
RECOMENDACIONES	65
BIBLIOGRAFÍA	66
GLOSARIO DE TÉRMINOS	70
ANEXOS	72
ANEXO 1. COMPARACIÓN ENTRE LAS METODOLOGÍAS ÁGILES Y LAS METODOLOGÍAS TRADICIONALES.	72

INTRODUCCIÓN

Según establece la Constitución de la República de Cuba, la Fiscalía General de la República es el órgano del Estado que tiene como objetivos fundamentales, el control y la preservación de la legalidad. Esta debe velar también porque los organismos, entidades económicas, sociales y ciudadanos, cumplan estrictamente la Constitución, las leyes y demás disposiciones legales; representando de esta forma al Estado en la promoción y el ejercicio de la acción penal pública.

Además de los objetivos planteados anteriormente, se le atribuyen los siguientes:

- ✓ Procurar el restablecimiento de la legalidad cuando sea quebrantada por disposiciones o decisiones contrarias a la Constitución y las leyes o por aplicación indebida o incumplimiento de éstas.
- ✓ Promover la sanción de quienes atenten contra la independencia y la soberanía del Estado, así como contra los intereses políticos, económicos y sociales de éste.
- ✓ Proteger a los ciudadanos en el ejercicio legítimo de sus derechos e intereses.
- ✓ Preservar los derechos e intereses legítimos de los órganos, instituciones y dependencias estatales y de las entidades económicas y sociales.
- ✓ Combatir toda manifestación de abuso de poder y corrupción.
- ✓ Contribuir a la prevención del delito y otras conductas antisociales, al fortalecimiento de la disciplina social y a la educación de los ciudadanos en la observancia consciente de las normas jurídicas.

La Fiscalía General de la República, constituye una unidad orgánica subordinada únicamente a la Asamblea Nacional del Poder Popular y al Consejo de Estado y recibe instrucciones directas de este último. A su vez, los órganos de la Fiscalía están organizados verticalmente en toda la nación, subordinados sólo a la Fiscalía General de la República y son independientes de todo órgano local. (Sitio Oficial Fiscalía General de la República)

Actualmente el trabajo en fiscalías del país presenta las siguientes deficiencias:

- Se realizan reportes estadísticos sobre datos de captación del personal de forma telefónica o mediante el correo.

- La gran cantidad de documentos que se generan diariamente son almacenados en archivos metálicos, por lo que las búsquedas de información se realizan de forma manual.
- Existe un excesivo consumo de papel que puede deteriorarse, extraviarse o causar demoras en la solución de los casos.

En todos los procesos descritos, es necesario invertir demasiado tiempo, recursos humanos y materiales, debido al gran volumen de trabajo existente, para poder tomar las decisiones más acertadas en el momento preciso. Los fiscales y el personal administrativo, necesitan informatizar el proceso de búsqueda de información, el registro de datos referentes al personal que labora y reportes que se generan en cada una de las fiscalías del país. De esta forma se lograría rapidez en la toma de decisiones, se elevaría la eficiencia laboral con la menor cantidad de recursos humanos y materiales posibles, y se alcanzaría una mayor conformidad por parte del cliente.

A raíz de esta problemática, la Fiscalía General de la República en convenio con la Universidad de Ciencias Informáticas, hacen nacer el proyecto Sistema de Gestión Fiscal (SGF). Este proyecto está compuesto por ocho subsistemas entre los que se encuentra Gestión de Cuadros y Personal de Apoyo. Este subsistema está formado por los módulos Personal de apoyo y Reportes y búsquedas, los cuales posibilitarán registrar y controlar al personal de apoyo que accederá a la aplicación, y realizar los reportes y búsquedas necesarios que permitan agilizar e informatizar el trabajo en las fiscalías.

Para lograr el correcto desarrollo de los módulos Personal de apoyo y Reportes y búsquedas, se necesita identificar las necesidades del cliente y traducirlas a un lenguaje que entiendan los desarrolladores. En la historia de la informática, muchos casos han evidenciado las incongruencias que han existido entre lo que el usuario quería, lo que realmente necesitaba, lo que interpretaba cada miembro del equipo de proyecto y lo que se obtiene realmente. Por esta razón se deben obtener requisitos que reflejen de forma clara y precisa qué debe y qué no debe hacer el sistema.

Teniendo en cuenta lo antes expuesto se plantea el siguiente **problema científico**: ¿Cómo realizar la captura de las necesidades del cliente en los módulos Personal de apoyo y Reportes y búsquedas del proyecto SGF para garantizar un entendimiento entre clientes y desarrolladores?

El **objeto de estudio** es la Ingeniería de requisitos y el **campo de acción** la elicitación, análisis, especificación y validación de los requisitos de los módulos Personal de apoyo y Reportes y búsquedas del proyecto SGF.

Se traza como **objetivo general** realizar la captura de las necesidades del cliente en los módulos Personal de apoyo y Reportes y búsquedas del proyecto SGF que garantice un entendimiento entre clientes y desarrolladores.

Hipótesis: Si se realiza la captura de las necesidades del cliente en los módulos Personal de apoyo y Reportes y búsquedas del proyecto SGF, entonces se garantizará un entendimiento entre clientes y desarrolladores.

Con el fin de dar solución al problema planteado y alcanzar el objetivo trazado se proponen las siguientes tareas de investigación:

1. Elaboración del marco teórico para la justificación del trabajo.
2. Identificación de los requisitos de software para obtener un listado de todas las necesidades del cliente.
3. Análisis de los requisitos de software con el fin de clasificarlos, eliminando todas las inconsistencias que los mismos puedan tener.
4. Especificación de los requisitos de software para presentarlos de manera consistente y comprensible.
5. Validación de los requisitos de software para evaluar la calidad del resultado de haber aplicado la elicitación, análisis y especificación de requisitos.

Métodos y técnicas de investigación a utilizar

Para validar metodológicamente la investigación utilizamos los siguientes métodos:

Teóricos

- **Histórico - Lógico:** Para el análisis de la trayectoria de la Ingeniería de Requisitos y para expresar teóricamente sus elementos fundamentales.
- **Hipotético - Deductivo:** Para la realización de la hipótesis.
- **Modelación:** Para la creación de modelos como abstracciones de la actualidad de los módulos, con el objetivo de realizar el análisis del proceso de desarrollo del sistema que dará solución a la problemática planteada.

Empíricos

- ✓ Entrevista: Para la obtención de información acerca de los requisitos que debe cumplir el software.

Estructura de la tesis

Capítulo 1. Se realiza la fundamentación teórica de la investigación con el estudio del estado del arte de sistemas relacionados con los módulos Personal de apoyo y Reportes y búsquedas. Se estudian además las metodologías de desarrollo de software, lenguajes de modelado, las herramientas CASE y de modelado de prototipos de interfaz de usuario. Se realiza un análisis de las principales etapas de elicitación, análisis, especificación, validación y gestión de la ingeniería de requisitos, así como sus técnicas; con el objeto de ver cuáles serán aplicadas en la solución del problema.

Capítulo 2. Se realiza la elicitación, análisis y la especificación de los requisitos de los módulos Personal de apoyo y Reportes y búsquedas. Como resultado de la aplicación de estas etapas, se identifican los requisitos funcionales y no funcionales del sistema y se confeccionan prototipos de interfaz que los validen.

Capítulo 3. Se realiza un análisis de los resultados obtenidos después de aplicar técnicas de validación y métricas orientadas a la calidad de las especificaciones de los requisitos obtenidas y de los diagramas de casos de usos.

CAPÍTULO 1. “FUNDAMENTACIÓN TEÓRICA”

1.1 Introducción

En este primer capítulo se describen los sistemas existentes en la actualidad detallándose sus principales características. Se caracterizan las metodologías de desarrollo de software, herramientas CASE y de modelado de la interfaz de usuario, así como de lenguajes de modelado. Se muestran además aspectos concernientes a la ingeniería de requisitos, las diferentes etapas que abarca y técnicas que facilitan su desarrollo.

1.2 Sistemas relacionados con los módulos Personal de apoyo y Reportes y búsquedas en el mundo

La informática hoy en día ha logrado un desarrollo vertiginoso, alcanzando ramas como el derecho, donde los adelantos tecnológicos obtenidos permiten que se brinde a los clientes un mejor servicio lleno de eficiencia y rapidez. Muchos son los sistemas que se han desarrollado en el mundo judicial para hacer más afable el trabajo y la atención brindada a la población, con el fin de conocer sus características a continuación se estudian alguno de ellos.

Gestión de Expedientes Jurídicos para Despachos de Abogados y Profesionales (Gedex)

Es un software sencillo que es utilizado para la gestión de expedientes en los despachos, bufetes o departamentos jurídicos. Garantiza que los abogados realicen un seguimiento completo de todos sus casos judiciales. Clasifica los expedientes enviados de forma instantánea, agilizando el proceso de búsqueda y la localización de casos. Tiene implementada una agenda de contactos que permite asociar rápidamente los expedientes a los mismos. Ofrece además un avanzado sistema de contraseñas, con el que puede limitar el acceso a la información, ocultar expedientes y contactos, y bloquear los cambios sin autorización. (AplicacionesEmpresariales.com; Gedex.net)

ABOGest

Está pensado para ser usado por abogados controlando sus asuntos judiciales, estados del trámite, citas pendientes y cuáles términos han llegado a su fin. Permite además agrupar expedientes, exportar la agenda de contactos a Microsoft Outlook, definir grupos de usuarios, adjuntar documentos a los

expedientes, ordenar los registros insertados, realizar búsquedas relacionadas con los expedientes, entre otros. (ABOGest: Novedades)

Abogafin

Abogafin permite ejercer un control total sobre los expedientes jurídicos, realizando un correcto seguimiento de los mismos. Gracias a su motor de búsqueda se puede localizar cualquier expediente de forma sencilla y ágil, así como acceder de forma inmediata a todo tipo de información y documentos asociados. Es un programa compatible con todas las versiones de Windows, que posibilita diseñar documentos en Word, PDF o Excel y adjuntarlos a los expedientes. Brinda un sistema de seguridad que garantiza el uso restringido a los datos, imposibilitando que usuarios sin privilegios ni derechos de acceso, puedan acceder al sistema. (ABOGAFIN software para abogados)

Sistema de información para la notificación telemática y la cooperación (LexNet)

Lexnet es un sistema de comunicación electrónica que interconecta a las diferentes oficinas judiciales que permite intercambiar documentos jurídicos de forma segura, garantizando que la Administración de Justicia realice su trabajo. Emplea notificaciones para informar al abogado respecto al seguimiento que se le da a los documentos, brindando un servicio rápido y con gran calidad. Está basado en un sistema de correo que garantiza el cumplimiento de principios fundamentales de la seguridad informática como la Integridad, Confidencialidad y Disponibilidad. (Lexnet)

Sistema de gestión fiscal (SGF)

Solución cubana que recoge los datos de los procesos relacionados con la protección de los derechos al ciudadano. Es una aplicación de escritorio desarrollada en el lenguaje de programación Visual Basic, sobre una base de datos hecha en Microsoft Access 2000. Inicialmente en este sistema, se pensó implementar un módulo que se encargaría de la gestión del personal de apoyo y de realizar reportes y búsquedas que agilizaran el trabajo diario que se lleva a cabo en las fiscalías del país, pero esta idea nunca se consolidó.

Conclusión parcial

Ninguno de los sistemas estudiados anteriormente, permiten la manipulación de datos relacionados con la información personal y curricular de los fiscales y de sus asistentes, ya que están pensados

para agilizar el control que ejercen los abogados sobre sus expedientes. Además, todos están desarrollados bajo licencias propietarias, frustrando el objetivo primordial que tiene el Estado cubano de migrar hacia plataformas de código abierto.

1.3 Proceso de desarrollo de software

El proceso de desarrollo de software "es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos 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" (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 (Proceso de Desarrollo de Software). Requiere un conjunto de conceptos, una metodología y un lenguaje propio. (La Ingeniería de Software)

1.4 Metodologías para el desarrollo del software

Un proceso de software detallado y completo suele denominarse "Metodología". Adicionalmente una metodología debería definir con precisión los artefactos, roles y actividades involucrados, junto con prácticas y técnicas recomendadas, guías de adaptación de la metodología al proyecto, guías para uso de herramientas de apoyo, etc. Habitualmente se utiliza el término "método" para referirse a técnicas, notaciones y guías asociadas, que son aplicables a una (o algunas) actividades del proceso de desarrollo.

La comparación y/o clasificación de metodologías no es una tarea sencilla debido a la diversidad de propuestas y diferencias en el grado de detalle, información disponible y alcance de cada una de ellas (Proceso de Desarrollo de Software). Estas se pueden dividir en dos categorías: Ágiles y Tradicionales.

1.4.1 Metodologías Ágiles

Estas metodologías surgen por la incapacidad que tienen las metodologías tradicionales de operar en condiciones volátiles. Se caracterizan por una gran reducción de los tiempos de desarrollo del software, planteando que es más importante la producción de un software funcional que procesar una

documentación excesiva. Poseen una alta capacidad de respuesta a cambios ya que en ellas no se sigue un plan estricto, lo que las hace más flexibles que las anteriores.

Las metodologías ágiles han presentado éxito en proyectos de poco alcance, donde el equipo de desarrollo no necesita ser amplio y se requiere la entrega inmediata del producto. Un ejemplo de ellas es eXtreme Programing (XP).

1.4.1.1 eXtreme Programing (XP)

La Programación Extrema es una metodología ligera de desarrollo de software basada en la simplicidad, ya que desarrolla y codifica los módulos del sistema, en la comunicación entre los clientes y desarrolladores, y la realimentación o reutilización del código desarrollado. Surge como respuesta y posible solución a los problemas derivados del cambio en los requisitos. En la mayoría de los casos se plantea como una metodología a emplear en los proyectos con riesgos de requisitos muy cambiantes.

Según Kent Beck «Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que se sabe que el cambio va a suceder; el problema es la incapacidad de adaptarse a dicho cambio cuando éste tiene lugar» (Escribano F. G., 2002).

Esta metodología es utilizada en proyectos de corto plazo, equipos pequeños y poco tiempo de entrega.

1.4.2 Metodologías Tradicionales

Estas metodologías se caracterizan por ser rígidas y dirigidas por la documentación que se genera en cada una de las actividades desarrolladas. Están basadas en la producción de proyectos de larga duración, lo que permite estructurar un amplio equipo de trabajo en roles que cumplirían diferentes funciones dentro de la producción.

La experiencia ha demostrado que las metodologías tradicionales no ofrecen una buena solución para proyectos donde el entorno es volátil y donde los requisitos no se conocen con exactitud, porque no están pensadas para trabajar con la incertidumbre. Entre algunas de estas metodologías se encuentran Rational Unified Process (RUP) y Microsoft Solution Framework (MSF).

1.4.2.1 Microsoft Solution Framework (MSF)

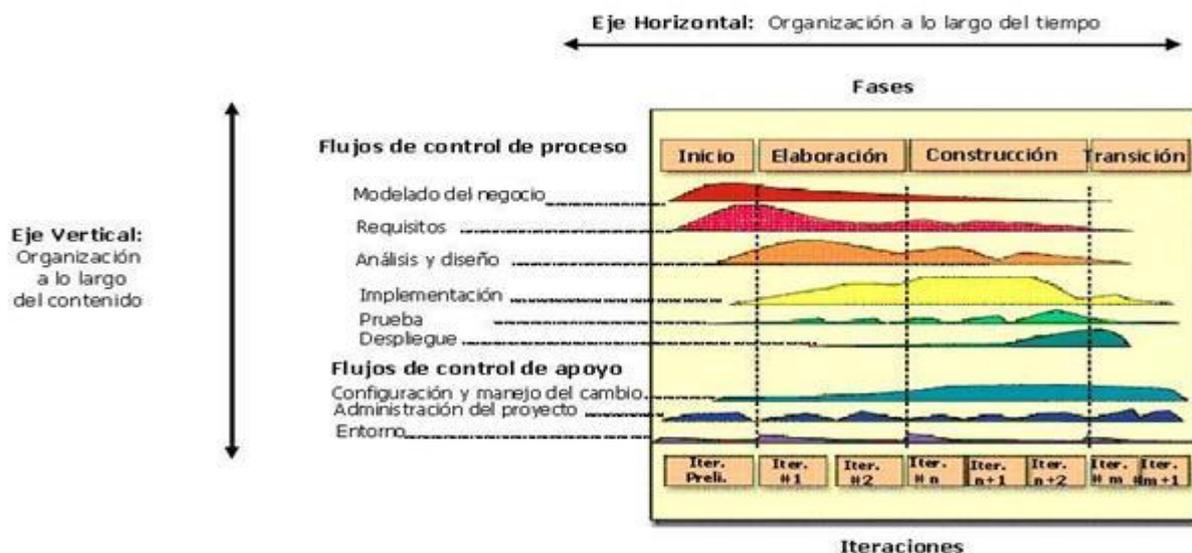
El Marco de solución de Microsoft, proporciona un sistema de modelos, principios, y pautas para dar soluciones a empresas que diseñan y desarrollan. Asegura que todos los elementos de un proyecto, tales como las personas, procesos, y herramientas, puedan ser manejados con éxito.

Esta es una metodología dócil y adaptable, que a través de conceptos, modelos y prácticas de uso, controla la planificación, el desarrollo y la gestión de proyectos tecnológicos. Se centra en los modelos de proceso y de equipo, dejando en un segundo plano las elecciones tecnológicas (Hurtado, 2006). Organiza equipos de 3 y 4 personas, así como proyectos de más de 50 personas. Se puede utilizar junto a cualquier tecnología o ambiente de desarrollo.

1.4.2.2 RUP (Rational Unified Process)

La siguiente figura muestra la arquitectura global de RUP y sus dos dimensiones:

- El eje horizontal representa el tiempo y muestra los aspectos del ciclo vital del proceso a medida que se desarrolla.
- El eje vertical representa las disciplinas que agrupan de forma lógica las actividades por naturaleza.



En esta figura se muestra cómo el énfasis va cambiando con el tiempo. Por ejemplo, en las primeras iteraciones, se empleaba más tiempo en los requisitos, mientras que en iteraciones posteriores se empleó más tiempo en la implementación (Ayuda de RUP). También se muestran las cuatro fases de desarrollo que abarca, cuyos objetivos se describen a continuación:

- Fase de Inicio: Determinar la visión del proyecto.
- Fase de Elaboración: Establecer la arquitectura óptima.
- Fase de Construcción: Obtener la capacidad operacional inicial.
- Fase de Transición: Desarrollar el release del proyecto. (Mendoza, 2004)

Características principales de RUP

1. Dirigido por casos de uso: Los casos de uso reflejan lo que los usuarios necesitan y desean, guiando así el proceso de desarrollo.
2. Iterativo e Incremental: Divide el proceso de desarrollo en varios mini-proyectos o versiones del producto donde a cada uno de estos se le denomina iteración y pueden o no representar un incremento en el grado de terminación del producto completo.
3. Centrado en la arquitectura: Describe los elementos del modelo que son más importantes para su construcción para mostrar mediante la arquitectura una visión común del sistema, en la que el equipo de proyecto y los usuarios están de acuerdo. (Jacobson, Booch, & Rumbaugh, 2000)

Comparación entre las metodologías ágiles y las metodologías tradicionales, ver Anexo 1.

Selección de la metodología de desarrollo de software a utilizar

Luego de la realización del estudio de algunas de las metodologías usadas en la actualidad, se puede concluir que RUP presenta características perfectamente ajustables al proyecto, por ejemplo: Ser iterativo e incremental es útil para proyectos de larga duración como es el caso de Sistema de Gestión Fiscal. Además, el equipo de desarrollo tiene experiencia en su aplicación, pues a pesar de estar compuesto por personal variable en el tiempo, la característica de que esta metodología genere gran

cantidad de documentación es necesaria; para apoyar el entendimiento de los procesos y casos de uso.

1.5 Herramientas CASE

La industria de computadoras ha desarrollado un soporte automatizado para el desarrollo y mantenimiento de software, llamado Computer Aided Software Engineering (CASE), que en su traducción al español significa: Ingeniería de Software Asistida por Computación. Se puede definir a las Herramientas CASE como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del Ciclo de Vida de desarrollo de un Software. Como es sabido, los estados en el Ciclo de Vida de desarrollo de un Software son: Investigación Preliminar, Análisis, Diseño, Implementación e Instalación (INEI).

Estas herramientas facilitan además la organización y manejo de la información de un proyecto informático. Permitiéndoles a los participantes de un proyecto, que los sistemas (especialmente los complejos), se tornen más flexibles, más comprensibles y además mejorar la comunicación entre los participantes (Case).

La principal ventaja de la utilización de una herramienta CASE, es la mejora de la calidad de los desarrollos realizados y, en segundo término, el aumento de la productividad. Para conseguir estos dos objetivos es conveniente contar con una organización y una metodología de trabajo, además de la propia herramienta (INEI).

1.5.1 Rational Rose

Permite representar gráficamente el sistema, haciendo énfasis en los detalles más importantes, centrándose en los casos de uso y enfocándose hacia un software de mayor calidad, empleando un lenguaje estándar común que facilita la comunicación.

Proporciona mecanismos para realizar la Ingeniería Inversa, es decir, a partir del código se puede obtener información sobre su diseño. Permite generar código en diferentes lenguajes a partir de un diseño en UML. Brinda la posibilidad de que varias personas trabajen a la vez, permitiendo que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo.

1.5.2 Visual Paradigm 6.4 for UML Enterprise Edition

Ofrece una excelente interoperabilidad con otras herramientas CASE y muchos de los entornos IDE líderes del mercado. Esta herramienta permite la captura de requisitos, análisis, diseño e implementación. También proporciona características tales como generación del código, ingeniería inversa y generación de informes. Incorpora el soporte para trabajo en equipo, permitiendo que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros. Soporta además, el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue; generando toda la documentación de lo que se hace cumpliendo con estándares establecidos.

Visual Paradigm es una de las pocas herramientas CASE que soporta el análisis textual. El análisis textual es una técnica útil y práctica para la captura de los requisitos del sistema y la identificación de las clases candidatas.

Selección de la herramienta CASE a utilizar

Se seleccionó Visual Paradigm, ya que esta es una herramienta multiplataforma de modelado visual UML muy fácil de utilizar. Esta característica es muy importante pues por ejemplo el Rational Rose, que es una herramienta muy recomendada y además profesional, tiene como desventaja; que obliga al usuario a trabajar en máquinas que tengan instalado el sistema operativo Windows. Por su parte, Visual Paradigm es una herramienta de gran utilidad para el proyecto, por su nivel de integración con el entorno de desarrollo Eclipse, y por la posibilidad que brinda de trabajar sobre la plataforma libre GNU/Linux, sistema operativo sobre el cual se desarrolla el proyecto.

1.6 Herramientas de modelado de prototipos de interfaz de usuario

Un prototipo es un modelo (representación, demostración o simulación) fácilmente ampliable y modificable de un sistema planificado, probablemente incluyendo su interfaz y su funcionalidad de entradas y salidas (Prototipado).

Los prototipos son utilizados ampliamente en el desarrollo de aplicaciones, para la evaluación de especificaciones de un sistema de información, o para un mejor entendimiento de cómo los requisitos de un sistema de información se ajustan a los objetivos perseguidos. Actualmente, es imprescindible

utilizar productos que incorporen esta funcionalidad por la cambiante tecnología y necesidades de los usuarios (INEI).

1.6.1 Microsoft Office Visio 2007

Office Visio 2007 facilita a los profesionales empresariales la visualización, análisis y comunicación de información, sistemas y procesos complejos. Los diagramas de aspecto profesional que incorpora, permiten mejorar la comprensión de sistemas y procesos, entender mejor la información compleja y utilizar esos conocimientos para tomar mejores decisiones para la empresa.

Facilita y agiliza la creación de una gran variedad de diagramas entre los que se encuentran: los diagramas de flujo de proceso empresarial, diagramas de red, diagramas de flujo de trabajo, modelos de bases de datos y diagramas de software, entre otros. Los diagramas están organizados en plantillas según su tipo. Permite además exportar a los formatos PDF y XML Paper Specification (XPS) para facilitar la portabilidad de los trabajos realizados (Microsoft Office Online).

1.6.2 Visual Paradigm for UML 6.4

En el epígrafe 1.5.2 se analizaron características funcionales de la herramienta Visual Paradigm como herramienta CASE para el desarrollo de los artefactos. Esta permite también la creación de prototipos de interfaz de usuario sencillos, para abundar más en las características del Visual Paradigm, ver el epígrafe 1.5.2.

1.6.3 Axure RP 5.5

Axure RP es una aplicación ideal para crear prototipos con especificaciones muy precisas para páginas web. Esta herramienta cuenta con todo lo que se puede necesitar para crear los prototipos de forma más eficiente. Permite componer la página web visualmente, añadiendo, quitando y modificando los elementos con suma facilidad.

Donde Axure RP demuestra su grado de especialización es en las anotaciones. En este punto, permite especificar el estado de cada elemento (Propuesto, Aceptado, Incorporado), el beneficio esperado (Crítico, Importante, Útil), el riesgo, la estabilidad, a quién va dirigido y a quién se le asignará la tarea.

Una de las principales limitaciones que presenta esta aplicación es que solo funciona en los siguientes sistemas operativos: Windows 98/98SE/Me/2000/XP (Gómez).

Entre las ventajas que provee están:

- ✓ Poseer una serie de objetos predefinidos que cubren la mayoría de las necesidades que puede tener el analista a la hora de integrar elementos en un prototipo.
- ✓ Facilitar la creación de plantillas.
- ✓ Cada elemento del prototipo puede ser comentado y definido y estos comentarios se pueden exportar a Word.
- ✓ Organización de prototipos en forma de árbol y exportación de los mismos a formato HTML y de imagen, facilitando la interacción entre prototipos y la presentación de los mismos a los clientes.

Selección de la herramienta de modelado de prototipos de interfaz de usuario a utilizar

Se selecciona el Axure RP ya que se quiere desarrollar una aplicación web y ésta, es una excelente herramienta para la confección de los prototipos de interfaz de usuario para este tipo de aplicaciones. Entre la variedad de facilidades que brinda se encuentran la generación de pantallas de forma rápida y sencilla y la interacción dinámica que se puede establecer entre cada una de ellas. Este software permite alcanzar una mejor percepción del funcionamiento futuro que tendrá el sistema posibilitando visualizar cómo quedará el sitio web que fue diseñado mediante prototipos, con solo presionar F5.

1.7 Lenguajes para el modelado

Un proceso de negocio es un conjunto de actividades relacionadas dentro de una organización que tienen como objetivo conseguir un determinado resultado. Los modelos de procesos de negocio se emplean para mejorar la comunicación tanto entre el analista y el desarrollador como entre el analista y el cliente (Pérez). Entre los lenguajes más usados, que permiten realizar modelos de procesos de negocio, están: IDEF y UML.

1.7.1 IDEF0

IDEF0 es una técnica de modelación concebida para representar de manera estructurada y jerárquica las actividades que conforman un sistema o empresa, y los objetos o datos que soportan la interacción de esas actividades. Se aplica para comunicar reglas y procesos de negocio, obtener una visión estratégica de un proceso y facilitar un análisis para identificar puntos de mejora (Pérez).

Entre las ventajas que provee están:

- ✓ Es una notación simple que cualquier cliente puede usar para describir qué hace en el negocio. Involucrar a los clientes de la organización en la modelación del negocio, permite ahorrar tiempo simultaneando el trabajo en varias áreas, así como obtener un modelo más fiel ya que ha sido elaborado por sus protagonistas.
- ✓ Permite descubrir problemas de organización en el negocio que deben ser arreglados, para "no informatizar el caos" sino organizar el negocio y luego informatizarlo. (Alonso Riverón, Cruz Navarro, & Tornés Medina)

1.7.2 BPMN

BPMN define un Diagrama de Procesos de Negocio (BPD, del inglés Business Process Diagram), que se basa en una técnica de grafos de flujo para crear modelos gráficos de operaciones de procesos de negocio. La creación de BPMN tuvo como base la recopilación de experiencias de varios estándares como UML e IDEF. Es importante tener en cuenta que abarca únicamente los procesos de negocio, lo que significa que otros tipos de modelos relacionados (estructura de la organización, recursos, modelos de datos, estrategias, reglas de negocio) quedan fuera de la especificación (Pérez).

Su principal objetivo es proporcionar una notación fácilmente comprensible por todos los usuarios del negocio, desde los analistas, los desarrolladores técnicos, hasta aquellos que monitorizarán y gestionarán los procesos.

Los diagramas BPMN, también llamados BPD están formados por una serie de elementos fundamentales. Estos se pueden clasificar en cuatro categorías fundamentales:

1. **Objetos de flujo:** Los 3 objetos de flujos principales son los Eventos (representan algo que ocurre durante el transcurso de un proceso de negocio), Actividades (cualquier trabajo que realiza) y Pasarelas (controlan el flujo).
2. **Conectores:** Son los elementos que servirán para conectar los diferentes Objetos de flujo con el objeto de crear el esqueleto estructural básico del proceso de negocio.
3. **Calles:** son el mecanismo que permite clasificar las actividades de manera visual para ilustrar las distintas categorías o responsabilidades.
4. **Artefactos:** Los 3 artefactos predefinidos son Datos (representan los datos que son producidos o requeridos por las actividades), Grupos (se emplean para agrupar distintos elementos del diagrama) y Anotaciones (para proporcionar información adicional).

1.7.3 Lenguaje de Modelado Unificado (UML)

UML es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad, que permite la modelación de sistemas con tecnología orientada a objetos. Se utiliza para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. Proporciona además una organización en el proceso de diseño de forma tal que analistas, clientes, desarrolladores y otras personas involucradas en el desarrollo del sistema lo comprendan y convengan con él (Schmuller J. , 2000).

Es un sistema de notación elegante, expresiva y flexible, que entre otras cosas incluye el significado de sus notaciones. UML no guía al desarrollador en la forma de realizar el análisis y diseño orientados a objetos, ni le indica cuál proceso de desarrollo adoptar (Larman, 1999). Un modelo UML puede ser usado en todas las fases de desarrollo de software, lo que facilita su entendimiento y la comunicación.

UML permite la implementación de diversos tipos de diagramas, dentro de los cuales se encuentra el diagrama de casos de uso, que modela la funcionalidad del sistema según lo perciben los usuarios externos, llamados actores.

Selección del lenguaje de modelado a utilizar

A pesar de ser UML el lenguaje de modelado que recomienda la metodología seleccionada (RUP) y de ofrecer una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas permitiendo modelar el ciclo completo de desarrollo del software; se decidió seleccionar dicho lenguaje debido a que el equipo de desarrollo tiene experiencia en su aplicación y por no disponer del tiempo necesario para la capacitación de otro lenguaje de modelado.

1.8 Ingeniería de Requisitos

El proceso de recopilar, analizar y verificar las necesidades del cliente para un sistema de software es llamado Ingeniería de Requisitos. Esta tiene como meta, entregar una especificación de requerimientos de software correcta y completa (Gil, 2002). Es un trabajo sistemático de desarrollo de requisitos, a través de un proceso iterativo y cooperativo de análisis del problema, documentando los resultados en una variedad de formatos, y probando la exactitud del conocimiento adquirido (Loucopoulos & Karakostas, 1995).

La Ingeniería de Requisitos cumple un papel primordial en el proceso de construcción y producción del software. Está basada en función de las necesidades planteadas por los clientes; definiendo el proceso a seguir, y facilitando la comprensión de lo que el cliente requiere. Permite además reconocer y establecer las funcionalidades que el sistema debe brindar, así como sus restricciones, logrando que tanto clientes como desarrolladores estén totalmente de acuerdo en lo que hay que hacer (¿Qué es Ingeniería de Requisitos?).

Para lograr un mejor entendimiento de lo que es la Ingeniería de Requisitos, es necesario definir qué son los requisitos, su importancia y clasificación.

Requisito de software

Los requisitos son la base fundamental de cualquier software que se desee desarrollar, ya que determinan las capacidades y cualidades que debe cumplir el sistema para garantizar una buena calidad. Una obtención de requisitos con calidad demuestra que el trabajo realizado culminará con éxito (Importancia de la captura de requerimientos en el proceso productivo del proyecto CISCOP de la facultad regional de Artemisa). Se pueden clasificar en Requisitos Funcionales y Requisitos No Funcionales.

Requisitos Funcionales:

Son aquellos que describen las funcionalidades del sistema; todas las actividades o servicios que debe realizar el software y que pueden ser comprobadas. (Jacobson, Booch, & Rumbaugh, 2000)

Requisitos No Funcionales:

Son los que limitan al hardware o software bajo el cual el sistema debe operar (Jacobson & Booch, 2004). Son propiedades o cualidades que debe tener el software. (Insfrán, Pastor, & Wieringa, 2002)

Los requisitos no funcionales pueden agruparse en las siguientes categorías:

- **Requisitos de Software:** Menciona el software que conviene tener instalado. (Jacobson, Booch, & Rumbaugh, 2000)
- **Requisitos de Hardware:** Elementos de hardware que se debe disponer. (Jacobson, Booch, & Rumbaugh, 2000)
- **Restricciones en el diseño y la implementación:** Especifica o restringe la codificación o construcción de un sistema. Son restricciones que han sido ordenadas y deben ser cumplidas estrictamente. (Jacobson, Booch, & Rumbaugh, 2000)
- **Requisitos de apariencia o interfaz externa:** Describe la apariencia del producto, especificando cómo se pretende que sea la interfaz externa del producto. Expresan el cumplimiento de normas o estándares de la empresa para la cual se esté desarrollando el software. (Jacobson, Booch, & Rumbaugh, 2000)
- **Requisitos de Seguridad:** Aseguran el cumplimiento de los fundamentos de la seguridad informática: Confidencialidad, Integridad y Disponibilidad. (Jacobson, Booch, & Rumbaugh, 2000)
- **Requisitos de Usabilidad:** Describen los niveles apropiados de usabilidad que se derivan de una combinación de lo que el cliente desea lograr con el producto y lo que los usuarios finales esperan del mismo (Jacobson, Booch, & Rumbaugh, 2000). Incluyen: factores humanos, estética, ayuda en línea, documentación de usuario y materiales de formación. (Ayuda de RUP)

- **Requisitos de Soporte:** Abarcan todas las acciones a tomar una vez que se ha terminado el desarrollo del software con motivos de asistir a los clientes para lograr el mejoramiento progresivo y evolución en el tiempo del software (Jacobson, Booch, & Rumbaugh, 2000). Incluyen: adaptabilidad, mantenimiento, compatibilidad, capacidad de configuración y la capacidad de instalación. (Ayuda de RUP)
- **Requisitos de Fiabilidad:** Estos requisitos incluyen la frecuencia y gravedad del error, la capacidad de recuperación, previsibilidad, precisión y el tiempo medio entre errores. (Ayuda de RUP)

1.8.1 Etapas de desarrollo de la ingeniería de requisitos

La Ingeniería de Requisitos se divide en etapas bien definidas que describen los pasos a seguir. El buen entendimiento y aplicación de cada una de estas etapas durante la primera fase del desarrollo del software, propicia una vía efectiva para mantener una fluida comunicación entre los involucrados en el desarrollo del proyecto, y una guía para especificar en detalles el sistema solicitado.

Elicitación

En esta etapa se lleva a cabo la captura o identificación de los requisitos. Este proceso se realiza cuando se ha determinado que el sistema a desarrollar es viable en costo y tiempo, contribuyendo de esta forma a los objetivos de la organización. Mediante esta actividad, el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema. (Escalona & Koch, 2002)

En la obtención de los requisitos juega un papel esencial el cliente, quien se convierte en un miembro más del equipo de proyecto por lo que se debe escribir en su lenguaje el resultado de la captura de los requisitos.

Análisis de requisitos

Una vez obtenidos los requisitos comienza entonces el proceso de análisis, clasificando los requisitos identificados y estudiándolos para eliminar todas las inconsistencias y ambigüedades presentes en los mismos. El análisis es una actividad iterativa donde los pasos del proceso deberán ser repetidos una cierta cantidad de veces, existiendo una constante comunicación entre la consulta de los clientes, los usuarios finales y los desarrolladores (Young, 2004).

En caso de existir algún tipo de conflicto con los requisitos identificados se debe solucionar mediante un proceso de negociación. Los riesgos asociados con cada requisito serán identificados y analizados. Se valora además el impacto de cada requisito con el coste y plazo de entrega del proyecto (Pressman, 2005).

En esta actividad se debe analizar la información recopilada, detectar puntos no claros y confirmar con el cliente todo lo comprendido. Es muy común en los usuarios solicitar más de lo que puede realizarse y proponer requisitos contradictorios.

Especificación de requisitos

El objetivo de esta etapa es obtener una especificación de los requisitos que ya han sido analizados y negociados con los clientes. El documento Especificación de Requisitos del Software (ERS) recoge de forma completa, precisa, y verificable, los requisitos que debe cumplir el sistema tanto funcional como no funcional.

Características que debe poseer una ERS:

Según la norma IEEE 830 – 1998 (IEEE'98b), una ERS debe ser:

- **Correcta:** la especificación contiene todos los requisitos que el software debe satisfacer. No hay una herramienta o procedimiento que garantice esta característica, sin embargo, limitarse a obtener de una vez lo que “quiere” el cliente pone en riesgo el logro de esta característica. Realizar un proceso iterativo, donde se involucren los interesados, seguir la traza de los requisitos y validarlos con los interesados contribuyen a lograr una ERS correcta. (IEEE'98a)
- **No ambigua:** cada requisito tiene una única interpretación. Para elaborar una ERS no ambigua es necesario mantener y documentar un acuerdo entre el equipo de desarrollo y los interesados respecto a los requisitos. (IEEE'98a)
- **Completa:** la especificación incluye los siguientes elementos:
 - ✓ Todos los requisitos funcionales y no funcionales son conocidos y documentados en la ERS.
 - ✓ En los documentos de la ERS todas las figuras, tablas, diagramas y definiciones de términos están nombrados y referenciados.

Desarrollar de forma iterativa la ERS de software, validarla y usar múltiples representaciones contribuyen a lograr la completitud. (IEEE'98a)

- **Consistente:** La ERS es consistente sí, y sólo sí, ningún subconjunto de la misma entra en contradicción con otro subconjunto (IEEE'98a). Las revisiones técnicas y el uso de diferentes representaciones para los requisitos contribuyen a reconocer la inconsistencia.
- **Clasificada por importancia:** La ERS cumple con esta característica si cada requisito puede ser identificado de manera única y tiene un atributo que indica su importancia desde el punto de vista de los interesados. Todos los requisitos no son igualmente importantes, algunos son críticos para el sistema y otros son deseables, conocer estos atributos es útil para planificar las iteraciones. (IEEE'98a)
- **Comprobable.** La ERS es comprobable sí, y sólo sí, se puede comprobar por una persona o máquina mediante un procedimiento finito y cuya relación costo-beneficio sea aceptable que cada requisito está en el sistema desarrollado (IEEE'98a).
- **Modificable.** La ERS es modificable sí, y sólo sí, puede ser modificada fácilmente manteniendo la completitud y consistencia. Mantener la traza de los requisitos, organizarlos adecuadamente y usar referencias cruzadas contribuye a hacerla modificable (IEEE'98a).
- **Posible de rastrear.** La ERS es posible de rastrear si está claro el origen de cada requisito y es posible determinar los elementos relacionados en etapas posteriores del desarrollo (IEEE'98a).

Validación de requisitos

Los requisitos una vez definidos necesitan ser validados. La validación de requisitos tiene como misión demostrar que la definición de los mismos define realmente el sistema que el usuario necesita o que el cliente desea (Lowe, Hall, & John, 1999). Es necesario asegurar que el análisis realizado y los resultados obtenidos en las etapas anteriores son correctos.

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 durante la validación detectar algunos problemas, y estos se deben corregir antes de aprobarse el documento final de requisitos.

Mediante la validación de especificaciones se asegura que el documento de requisitos represente una descripción clara del sistema, y que los requisitos cubren las necesidades de los usuarios.

Gestión de requisitos

La gestión de los requisitos es un conjunto de actividades que ayudan al equipo de proyecto a identificar, controlar y rastrear los requisitos y los cambios a éstos en cualquier momento mientras se desarrolla el proyecto (Pressman, 2005).

Con la aplicación de técnicas de validación y las métricas establecidas para el levantamiento de los requisitos de los módulos Personal de apoyo y Reportes y búsquedas, se aseguró la detección y corrección de los cambios necesarios en los mismos; por lo que no se consideró necesario aplicar aspectos más complejos de la gestión de requisitos como parte del alcance del presente trabajo, aunque se sugiera al proyecto global su implementación.

1.8.2 Técnicas de la ingeniería de requisitos

Existen un gran número de técnicas que se emplean en todo el mundo para realizar la captura y validación de los requisitos de un sistema. El empleo de las mismas garantiza que las etapas de la ingeniería de requisitos se cumplan con mayor calidad. A continuación se describirán algunas de estas técnicas.

1.8.2.1 Técnicas para la captura de requisitos

- **Entrevistas:** consisten en establecer un canal de comunicación directo entre los clientes y el equipo de desarrollo. Las entrevistas son una forma de conversación, no de interrogación, y se pueden realizar sobre la base de un cuestionario rígido o de una guía más o menos detallada que las orienta hacia puntos bien definidos.

Es la técnica de elicitación más utilizada, y de hecho es prácticamente inevitable en cualquier desarrollo. En las entrevistas se pueden identificar claramente tres fases: preparación, realización y análisis (Piattini, Calvo-Manzano, Cervera, & Fernández, 1996), que se describen a continuación:

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

Estudiar el dominio del problema: se debe conocer la terminología básica del dominio del problema, evitando que el cliente tenga que explicar términos que para él son obvios. Para ello se puede recurrir a la técnica auxiliar de estudio de documentación, a bibliografía sobre el tema, documentación de proyectos similares realizados anteriormente, etc.

Seleccionar a las personas a las que se va a entrevistar: se debe minimizar el número de entrevistas a realizar, por lo que es fundamental seleccionar a las personas a entrevistar. El orden de realización de las entrevistas también es importante. Normalmente se aplica un enfoque *top-down*, comenzando por los directivos, que pueden ofrecer una visión global, ayudar a determinar los objetivos y reducir ciertas *reticencias* en sus subordinados, y terminando por los futuros usuarios, que pueden aportar información más detallada.

Determinar el objetivo y contenido de las entrevistas: para minimizar el tiempo de la entrevista es fundamental fijar el objetivo que se pretende alcanzar y determinar previamente su contenido.

Planificar las entrevistas: la fecha, hora, lugar y duración de las entrevista deben fijarse teniendo en cuenta siempre la agenda del entrevistado.

2. Realización: Dentro de la realización de las entrevistas se distinguen tres etapas:

Apertura: el entrevistador debe presentarse e informar al entrevistado sobre la razón de la entrevista, qué se espera conseguir, cómo se utilizará la información, la mecánica de las preguntas, etc.

Desarrollo: la entrevista en sí no debería durar más de dos horas, distribuyendo el tiempo en un 20% para el entrevistador y un 80% para el entrevistado. Se deben evitar los monólogos y mantener el control por parte del entrevistador, contemplando la posibilidad de que una tercera persona tome notas durante la entrevista o grabar la entrevista en cinta de vídeo o audio, siempre que el entrevistado esté de acuerdo.

Terminación: se debe recapitular sobre la entrevista para confirmar que no ha habido confusiones en la información recogida, agradecer al entrevistado su colaboración y citarles para una nueva entrevista si fuera necesario, dejando siempre abierta la posibilidad de volver a contactar para aclarar dudas que surjan al estudiar la información o al contrastarla con otros entrevistados.

3. Análisis: Una vez realizada la entrevista es necesario leer las notas tomadas, pasarlas a limpio, reorganizar la información, contrastarla con otras entrevistas o fuentes de información, etc. Una vez elaborada la información, se puede enviar al entrevistado para confirmar los contenidos. También es importante evaluar la propia entrevista para determinar los aspectos mejorables.

- **Cuestionario:** consiste en un conjunto de preguntas presentadas a un grupo de personas para su respuesta. La forma de la pregunta puede influir en las respuestas, por lo que hay que planearlas cuidadosamente.

Las preguntas suelen distinguirse en dos categorías: abiertas y cerradas. Las preguntas abiertas permiten que los encuestados respondan con su propia terminología. Son especialmente útiles en la etapa exploratoria de la investigación, cuando se busca penetrar en el pensamiento del encuestado.

Desventajas de los cuestionarios:

- Muchos usuarios que pueden ofrecer una buena cantidad de información de forma oral, a la hora de responder los cuestionarios escriben poco o se resisten.
- Los entrevistados pueden objetar muchas preguntas, interpretarlas a su forma o no tomarlas en serio.
- Es difícil diseñar cuestionarios que aseguren obtener exactamente toda la información que se desea.

- **Sistemas existentes:** esta técnica consiste en analizar distintos sistemas ya desarrollados que estén relacionados con el sistema a ser construido. Esta técnica no necesita de la intervención del cliente, ya que mediante Internet se pueden buscar demos de productos que resultan similares y establecer contactos con profesionales que desarrollan sistemas de características comparables, aunque esto requiere de una profunda investigación y análisis.

Es recomendable que se le muestre al cliente el sistema luego de haberlo analizado, ya que por su experiencia puede sugerir importantes ideas nuevas.

- **Tormenta de ideas:** esta técnica permite generar gran cantidad de ideas en breve tiempo. Se desarrolla con un grupo de expertos a los que se les expone un problema, o se les envía un memorándum previo. Las ideas se generan y exponen por los asistentes de forma clara y precisa, evitando discursos, sin que medie ninguna crítica o evaluación de éstas, por descabelladas que pudieran parecer.

En esta atmósfera no crítica, las personas se sienten libres para decir lo que piensan y estas ideas, aún en el caso de que no tuvieran valor, pueden dar origen a otras por asociación. Las ideas se recogen y listan en papeles que se mantienen a la vista de todos para ser valoradas posteriormente.

- **Grabaciones de video y de 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, son importantes ya que permiten centrar la atención en la entrevista en sí, en vez de distraerse tomando notas de todo lo que se dice.

Esta técnica permite analizar los temas con más detenimiento y con una visión más global. Brinda la posibilidad de ver y analizar en detalle ese proceso la cantidad de veces que sea necesario.

- **Casos de Uso:** Aunque inicialmente se desarrollaron como técnica para la definición de requisitos (Jacobson I. , 1995), algunos autores proponen casos de uso como técnica para la captura de requisitos (Pan, Zhu, & Johnson, 2001). Un caso de uso describe la secuencia de interacciones que se producen entre el sistema y los actores del mismo para realizar una determinada función.

La ventaja esencial de los casos de uso es que resultan muy fáciles de entender para el usuario o cliente, sin embargo carecen de la precisión necesaria (Vilain, Schwabe, & Sieckenius, 2000) si no se acompañan con una información textual o detallada, y con otra técnica como pueden ser los diagramas de actividades. (Insfrán, Pastor, & Wieringa, 2002)

1.8.2.2 Técnicas de validación de requisitos

- **Revisiones:** Está técnica consiste en la lectura y corrección de la completa documentación o modelado de la definición de requisitos. Con la aplicación de esta técnica solo se puede validar la correcta interpretación de la información transmitida.

- **Auditorías:** Consiste en un chequeo de los resultados contra una lista de chequeo predefinida o definida a comienzos del proceso, es decir, se chequea cada requisito con ayuda de un cuestionario.

Las siguientes preguntas representan un pequeño subconjunto de las que pueden plantearse:

- ¿Está el requisito claramente definido?, ¿Puede interpretarse mal?

- ¿Está identificado el origen del requisito, (por ejemplo: persona, norma, documento)?
- ¿Qué otros requisitos hacen referencia al requisito estudiado?, ¿Están claramente identificados por medio de una matriz de referencias cruzadas o por cualquier otro mecanismo?
- ¿El requisito incumple alguna restricción definida?
- ¿Se puede localizar el requisito en el conjunto de objetivos del sistema?
- ¿El requisito es verificable?, Si es así, ¿Podemos efectuar pruebas para verificar el requisito?

Las preguntas planteadas en la lista de comprobación ayudan a asegurar que el equipo de validación dispone de lo necesario para realizar una revisión completa de cada requisito.

- **Matrices de trazabilidad:** Esta técnica consiste en marcar los objetivos del sistema y chequearlos contra los requisitos del mismo (Bernárdez, Durán, & Toro, 2004). Es necesario ir viendo qué objetivos cubre cada requisito, de esta forma se podrán detectar inconsistencias u objetivos no cubiertos.

- **Prototipos:** Algunas propuestas se basan en obtener de la definición de requisitos, prototipos que, sin tener la totalidad de la funcionalidad del sistema, permitan al usuario hacerse una idea de la estructura de la interfaz del sistema con el usuario. (Olsina, 1999)

Esta técnica presenta como problema que el usuario debe entender que lo que está viendo es un prototipo y no el sistema final.

1.9 Conclusiones parciales

El estudio realizado sobre los sistemas jurídicos existentes en el mundo, demostró que actualmente, ninguno se ajusta a las necesidades que tiene la Fiscalía General de la República de Cuba. Resultó la selección de la metodología RUP para guiar el desarrollo de un software con calidad, el Lenguaje Unificado de Modelado para mediante un lenguaje común facilitar el entendimiento entre todos los involucrados, la herramienta CASE Visual Paradigm 6.4 para facilitar la organización y manejo de toda la información que se genera en el proyecto y la herramienta Axure RP 5.5 para crear prototipos web de aspecto profesional.

CAPÍTULO 2. “SOLUCIÓN PROPUESTA”

2.1 Introducción

En este capítulo se describen las actividades que guían el proceso de levantamiento de requisitos de los módulos Personal de apoyo y Reportes y búsquedas del subsistema Gestión de Cuadros y Personal de Apoyo. Se modela el negocio como forma de garantizar una mayor comprensión de los principales procesos que se llevan a cabo en la Fiscalía General de la República; identificando los Actores del negocio, Trabajadores, Casos de uso, Reglas del negocio, Modelo de objetos y la Realización de los casos de uso del negocio; artefactos que propone la metodología tradicional escogida: RUP, para el flujo de trabajo Modelado de Negocio. También se especifican los requisitos funcionales y no funcionales que debe presentar el sistema que será construido, y se determinan los Casos de uso del sistema, los Actores que interactuarán con éste y la realización de los casos de uso.

2.2 Planificación del levantamiento de requisitos

Como premisas se guía la planificación por lo que establece la metodología tradicional escogida, RUP, y por los principios establecidos en el Libro de Proceso para la Administración de Requisitos del programa de mejoras de la UCI.

Atendiendo a las características del proyecto SGF, se elabora una guía para llevar a cabo el levantamiento de requisitos, que va estar compuesta por las diferentes etapas de la Ingeniería de requisitos (Elicitación, Análisis, Especificación y Validación). En estas etapas se realizarán un conjunto de tareas y se definirán las técnicas de la ingeniería de requisitos que se emplearán. El objetivo principal de esta guía es garantizar que el proceso de levantamiento sea llevado de manera uniforme, logrando obtener todos los requisitos que se encuentran en concordancia con lo desea el cliente.

A continuación se describen las actividades y tareas que se llevarán a cabo para levantar los requisitos de los módulos Personal de apoyo y Reportes y búsquedas.

Etapas inicial

Se realiza una reunión inicial (con los clientes y el grupo de trabajo), que tiene como objetivo establecer los elementos que se persiguen con el levantamiento de requisitos. Se explican cuáles son

las necesidades del cliente y se muestra la planificación que establece el proyecto. Se define además, flujo, cronograma con las actividades que se planifican (tanto con los clientes como con las personas que integran el Equipo de Desarrollo) y cuáles son los responsables del trabajo (tanto por la parte funcional como por del cliente).

Para el intercambio necesario entre clientes y desarrolladores, la entidad cliente pone a disposición del proyecto un Especialista por cada uno de los módulos que se desean implementar, que cumple con los siguientes requisitos:

- Conocimiento del manejo del negocio.
- Disponibilidad de tiempo.
- Grado de autoridad que tiene para la toma de decisiones.
- Compromiso con el proyecto.
- Habilidades de comunicación.
- Conocimientos de informática, entre otras.

En esta reunión se toma como acuerdo que los Especialistas facilitarán la documentación, procedimientos y normas necesarios para cumplir efectivamente el Plan establecido. También deben participar activamente en la captura y validación de los requisitos.

Responsables de la actividad: Jefe de Proyecto, Planificador y Analista.

Etapas de elicitación

El objetivo de esta etapa es obtener una lista de los requisitos que expresan las necesidades del cliente, en este proceso interviene el Analista y Especialista.

Tareas que se llevarán a cabo en la etapa:

- Recibir y analizar la documentación del negocio brindada por el Especialista, definido en la Reunión de inicio, de los procesos elementales del negocio.
- Aplicar las técnicas para la captura de requisitos: Se realizan Entrevistas al Especialista, se identificarán y describirán los Casos de uso. La aplicación de estas técnicas ayuda a obtener las necesidades fundamentales que el sistema deberá satisfacer y sus restricciones.

- Confeccionar un modelo de casos de uso que identifique los actores y los casos de uso del negocio. Este modelo de casos de uso permite a los modeladores comprender mejor qué valor proporciona el negocio a sus actores.
- Realizar los diagramas de actividades de los casos de uso del negocio para un mejor entendimiento del cliente.
- Desarrollar un modelo de objetos del negocio compuesto por trabajadores y entidades del negocio y sus relaciones.
- Definir las reglas del negocio asociadas al campo de acción que se modela.
- Realizar la captura de los requisitos: Con los artefactos generados hasta el momento y la participación activa del Especialista, se podrán extraer los requisitos que deberá cumplir el futuro sistema.

Etapas de Análisis

Esta etapa tiene como objetivo analizar los requisitos extraídos en la etapa anterior y eliminar todas las contradicciones que se detecten, en este proceso interviene el Analista y Especialista.

Tareas que se llevarán a cabo en la etapa:

- Clasificar y priorizar los requisitos: Se clasificarán los requerimientos identificados y se eliminarán todas las inconsistencias y ambigüedades que los mismos poseen. Para poder determinar qué casos de uso se desarrollarán en cada release de construcción. RUP propone clasificarlos de acuerdo al impacto que tienen en la arquitectura en:

- Críticos: Más importantes para los usuarios porque cubren las principales tareas o funciones que el sistema ha de realizar. Definen la arquitectura básica.

- Secundarios: Sirven de apoyo a los casos de uso críticos, involucran funciones secundarias y tienen un impacto más modesto sobre la arquitectura, pero deben implementarse pronto porque responden a requerimientos de interés para los usuarios.

- Auxiliares: No son claves para la arquitectura y completan casos de uso críticos o secundarios.

- Opcionales: Responden a funcionalidades que pueden o no estar en la aplicación, pero que no son imprescindibles en las primeras versiones.

- Realizar una reunión de conclusión del análisis con el Especialista, para darle a conocer y analizar los resultados obtenidos, dejando bien definidos los requisitos capturados.

Etapa de Especificación de requisitos

El objetivo de esta etapa es obtener el documento de Especificación de los Requisitos del Software que contiene detallados todos los requisitos que el software debe satisfacer, tanto funcionales como no funcionales. Contiene además todos los diagramas, figuras, tablas y definiciones de términos, en este proceso interviene el Analista.

Tareas que se llevarán a cabo en la etapa:

- Describir detalladamente los requisitos funcionales y no funcionales.

- Obtener el Modelo de caso de usos del sistema que permitirá identificar actores, casos de uso del sistema y sus relaciones.

- Especificar los casos de uso del sistema: Esta actividad permitirá describir el comportamiento del sistema ante las acciones que realizará el usuario.

- Elaborar prototipos de interfaz de usuario que representen las funcionalidades básicas que debe cumplir el sistema, identificando las pautas de diseño que tendrá la aplicación y las entradas y las salidas de los datos en la misma.

Etapa de Validación de los requisitos

El objetivo de esta etapa es asegurar la calidad de las etapas anteriores; eliminando errores de contenido, de interpretación, inconsistencias, requisitos contradictorios o imposibles de cumplir. En este proceso intervienen todos los involucrados.

Tareas que se llevarán a cabo en la etapa:

- Revisar que los prototipos diseñados recogen todos los requisitos identificados. Para el cumplimiento de esta tarea, se usará como ayuda el documento Especificación de requisitos de los módulos Personal de apoyo y Reportes y búsquedas y las descripciones de casos de uso.
- Aplicar las siguientes técnicas de validación:
 - Matrices de trazabilidad. Permitirán validar los requisitos contra los criterios del cliente.
 - Auditorías. Se aplicarán listas de chequeo para obtener los requisitos válidos y aquellos que no cumplen los criterios de validación. Se revisarán 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.
 - Revisiones. Permitirán analizar la documentación obtenida mediante la lectura, para de esta forma verificar si la interpretación de los documentos generados es correcta.
- Aplicar métricas que validen la calidad alcanzada en las tareas y actividades realizadas hasta el momento.
- Refinar requisitos: Esta actividad permitirá realizar las modificaciones pertinentes para corregir todos los errores encontrados mediante la aplicación de las técnicas descritas anteriormente (proceso de corrección de errores).

2.3 Modelado del Negocio del módulo Personal de apoyo

El Modelo del Negocio es un modelo de casos de uso que describe los procesos del negocio en términos de casos de uso y actores. Entre sus principales objetivos están:

- ✓ Comprender la estructura y la dinámica, problemas actuales e identificar las mejoras potenciales de la organización en la cual se va a desarrollar el sistema.
- ✓ Asegurar que los clientes, usuarios finales y desarrolladores, tengan un entendimiento común de la organización.
- ✓ Derivar los requerimientos del sistema que va a soportar la organización.

Como forma de cumplir con dichos objetivos, el proceso de modelado permite obtener una visión que refleja los procesos, roles y responsabilidades de la organización en los modelos de casos de uso del negocio y de objetos.

2.3.1 Reglas del negocio

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, regulando algún aspecto del negocio. Las reglas del negocio identificadas son:

- ✓ No se puede violar el término de los trámites establecidos en los procesos que se realizan en la fiscalía.
- ✓ Todos los controles existentes en la Fiscalía deben de estar actualizados.
- ✓ El investigador tiene 90 días hábiles para comprobar la conducta y moralidad del aspirante a una determinada plaza laboral.
- ✓ El Curso de Habilitación para el aspirante al cargo de Asistente del Fiscal en la FGR se impartirá por un período de 3 meses.
- ✓ El nombramiento del Asistente del Fiscal, se formalizará mediante la resolución que suscribe el Fiscal General de la República para la Fiscalía General, y para los correspondientes Fiscales Jefes Provinciales.
- ✓ La Dirección de Cuadros y Capacitación elaborará y controlará la aplicación del programa del Curso de Habilitación del Asistente del Fiscal.
- ✓ El Fiscal Jefe de la Dirección de Cuadros y Capacitación y el Director de Administración, tienen la facultad para dictar disposiciones complementarias que se requieran para la aplicación de lo dispuesto.

2.3.2 Actores del negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; que interactúa con el negocio para beneficiarse de los procesos desarrollados en el mismo. En el negocio se identifican los siguientes roles:

Actor	Descripción
Interesado	Es la persona que se presenta a la Fiscalía General enviado por el MTSS, resultado de captaciones realizadas por un profesor de la universidad en la que recibe clases o es un trabajador de la fiscalía. Puede ser un aspirante al cargo de asistente del fiscal o un aspirante a la plaza X (dígase plaza de Secretaria, Oficinista, Informático o

	Personal de servicio).
--	------------------------

2.3.3 Trabajadores del negocio

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades. Manipula entidades del negocio y puede interactuar con otros trabajadores. En el negocio se identifican los siguientes roles:

Trabajador	Descripción
Jefe de la Dirección de Administración	Es la persona encargada de controlar a todos los datos personales o laborales de los recursos humanos que trabajan en la fiscalía.
Secretaria de Administración	Asiste al Jefe de la Dirección de Administración en el cumplimiento de sus tareas.
Investigador	Persona encargada de realizar investigaciones de rigor que comprueben la conducta y moral del interesado a ocupar plaza en la fiscalía.
Jefe de la Dirección de Cuadros y Capacitación	Persona encargada de organizar y llevar a cabo el Curso de Habilitación para ocupar la plaza de asistente del fiscal, así como sus evaluaciones.
Fiscal General	Máxima autoridad encargada de revisar la propuesta de nombramiento dada a conocer por el Jefe de la Dirección de Cuadros y Capacitación y de aprobarla, mediante la Resolución de nombramiento.

2.3.4 Casos de uso del negocio

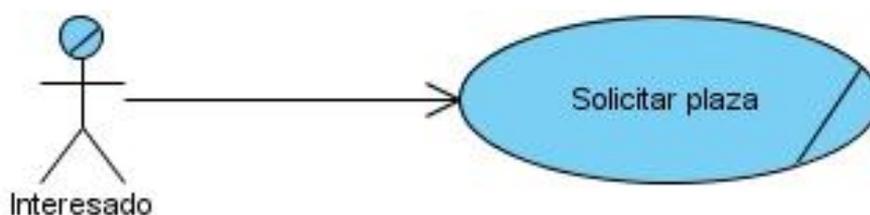
Un caso de uso del negocio representa un proceso del negocio, es decir, es una secuencia de actividades que producen un resultado observable para ciertos actores del negocio. Desde la perspectiva de un actor individual, define un flujo de trabajo completo que produce resultados deseables.

El caso de uso identificado en el negocio “Solicitar plaza”, se detalla en el epígrafe **2.3.6 Realización de casos de uso del negocio**.

2.3.5 Diagrama de casos de uso del negocio

Diagrama de caso de uso del negocio del módulo Personal de apoyo

Un diagrama de casos de uso del negocio representa gráficamente a los procesos del negocio y su interacción con los actores del negocio.



2.3.6 Realización de casos de uso del negocio

A continuación se expone la realización de los casos de uso donde se detallan los procesos del negocio, dejando bien claro cómo colaboran los trabajadores y las entidades del negocio en la ejecución del proceso.

Descripción textual Caso de Uso del Negocio: Solicitar plaza

Actores:	Interesado (inicia)
Trabajadores:	Jefe de la Dirección de Administración, Investigador, Secretaria de Administración, Jefe de Dirección de Cuadros y Capacitación, Fiscal General.
Resumen:	El caso de uso se inicia cuando el aspirante a una plaza laboral se presenta a la fiscalía con los documentos necesarios para adquirir la misma. Como resultado el investigador realiza las comprobaciones necesarias para comprobar la conducta y moralidad del interesado. El proceso termina cuando el Jefe de la Dirección de Administración le otorga el cargo al Interesado o Fiscal General aprueba la plaza de asistente mediante la Resolución de nombramiento.

Casos de uso asociados:	
Precondición:	
Flujo Básico	
Acción del Actor	Respuesta del Negocio
1- El Interesado se presenta en la Fiscalía General.	2- El Jefe de la Dirección de Administración le pide los siguientes documentos: <ul style="list-style-type: none"> - Título original del nivel escolar. - Fotocopia de dicho documento. - Autobiografía. - Tres fotos carné. - Certificado médico pre empleo. - Anexo 1 del comité militar (situación en la defensa).
	3- El Jefe de la Dirección de Administración verifica que los documentos entregados son legítimos. Si no son legítimos ir al Flujo Alternativo al Paso 3.
	4- El Jefe de la Dirección de Administración le entrega al Interesado el Modelo de entrevista.
5- El Interesado llena el modelo entregado.	6- La Secretaria de Administración comprueba si existen vacantes para la plaza solicitada, consultando el Modelo de plazas ocupadas. Si no existen plazas vacantes para la solicitud realizada ir al Flujo Alternativo al Paso 6. 7- El Investigador realiza investigaciones de rigor para comprobar la conducta y moralidad del Interesado, dejando constancia escrita en el documento "Resultado de la investigación", según Resolución No. 8 del reglamento del trabajo. Si las investigaciones realizadas no son satisfactorias ir al Flujo Alternativo al Paso 7.

	<p>7.1 Si la persona está optando por el cargo de asistente del fiscal, ir al paso 8.</p> <p>7.2 Si la persona está optando por el cargo X, ir al paso 11.</p> <p>8- El Jefe de la Dirección de Administración conforma un expediente con los documentos que entregó el Interesado y Resultado de la investigación.</p> <p>9- El Jefe de la Dirección de Administración entrega el expediente al Jefe de la Dirección de Cuadros y Capacitación. Ir a la Sección 1 “Realizar Curso de Habilitación”.</p> <p>11- Se le otorga el cargo X al Interesado.</p> <ul style="list-style-type: none"> • Termina el caso de uso.
Flujo Alternativo al Paso 3. “Los documentos entregados no son legítimos”	
	<p>3a) El Jefe de la Dirección de Administración le informa al Interesado los motivos por los que no puede ocupar la plaza.</p> <ul style="list-style-type: none"> • Termina el caso de uso.
Flujo Alternativo al Paso 6. “No existen plazas vacantes para la solicitud realizada”	
	<p>6a) La Secretaria de Administración le informa al Interesado que no existen plazas vacantes para la plaza solicitada.</p> <p>6b) La Secretaria de Administración guarda los documentos entregados en un archivo metálico.</p> <ul style="list-style-type: none"> • Termina el caso de uso.
Flujo Alternativo al Paso 7. “Las investigaciones realizadas no son satisfactorias”	
	<p>7a) La Secretaria de Administración cita al Interesado.</p>
7b) El Interesado se presenta en la fiscalía el día acordado.	<p>7c) El Jefe de la Dirección de Administración le informa al Interesado que la petición de la plaza solicitada fue rechazada y le devuelve la</p>

	<p>documentación que en su momento entregó.</p> <ul style="list-style-type: none"> • Termina el caso de uso.
Sección 1: “Realizar Curso de Habilitación”	
Acción del Actor	Respuesta del Negocio
	<p>1- El Jefe de Dirección de Cuadros y Capacitación recepciona los documentos entregados por el Jefe de la Dirección de Administración.</p> <p>2- El Jefe de Dirección de Cuadros y Capacitación organiza el Curso de Habilitación, junto a las unidades organizativas donde será ubicado el Interesado.</p> <p>3- El Jefe de Dirección de Cuadros y Capacitación le comunica al Jefe de la Dirección de Administración la fecha de inicio del curso.</p> <p>4- La Secretaria de Administración cita al Interesado.</p>
5- El Interesado se presenta en la fiscalía el día acordado.	6- El Jefe de la Dirección de Administración formaliza el contrato de trabajo por un periodo de 3 meses (tiempo que dura el Curso de Habilitación).
7- El Interesado aprueba el programa de estudio oficialmente establecido para el Curso de Habilitación. Si no aprueba el programa de estudio ir al Flujo Alterno al Paso 7 .	<p>8- El Jefe de Dirección de Cuadros y Capacitación eleva la propuesta de nombramiento del asistente al Fiscal General mediante el envío del expediente y las notas obtenidas en el curso.</p> <p>9- El Fiscal General revisa documentación y aprueba el nombramiento. Si no aprueba el nombramiento ir al Flujo Alterno al Paso 9.</p> <p>10- El Fiscal General redacta la Resolución de nombramiento del asistente del fiscal.</p>

	<p>12- El Jefe de Dirección de Cuadros y Capacitación recibe y saca copia de la resolución.</p> <p>13- El Jefe de la Dirección de Administración recibe copia de la resolución y formaliza la relación laboral como funcionario de la Fiscalía General de la República.</p> <p>14- La Secretaria de Administración llena los documentos: EP1, EP2 y EP3.</p> <p>15- La Secretaria de Administración le informa al Interesado de la aprobación de su nombramiento.</p> <ul style="list-style-type: none"> • Termina el caso de uso.
Flujo Alterno al Paso 7. “Desaprueba el Curso de Habilitación”	
	7a) La Secretaria de Administración cita al Interesado.
7b) El Interesado se presenta en la fiscalía el día acordado.	7c) El Jefe de administración le informa que su petición fue rechazada. <ul style="list-style-type: none"> • Termina el caso de uso.
Flujos Alternos al Paso 9. “Fiscal General no aprueba el nombramiento”	
	<p>9a) El Fiscal General le comunica al Jefe de Dirección de Cuadros y Capacitación que rechaza la propuesta de nombramiento.</p> <p>9b) El Jefe de Dirección de Cuadros y Capacitación le informa al Jefe de la Dirección de Administración que la propuesta de nombramiento fue rechazada.</p> <p>9c) La Secretaria de Administración cita al Interesado.</p>
9d) El Interesado se presenta en la fiscalía el día acordado.	9e) El Jefe de administración le informa que su petición fue rechazada. <ul style="list-style-type: none"> • Termina el caso de uso.

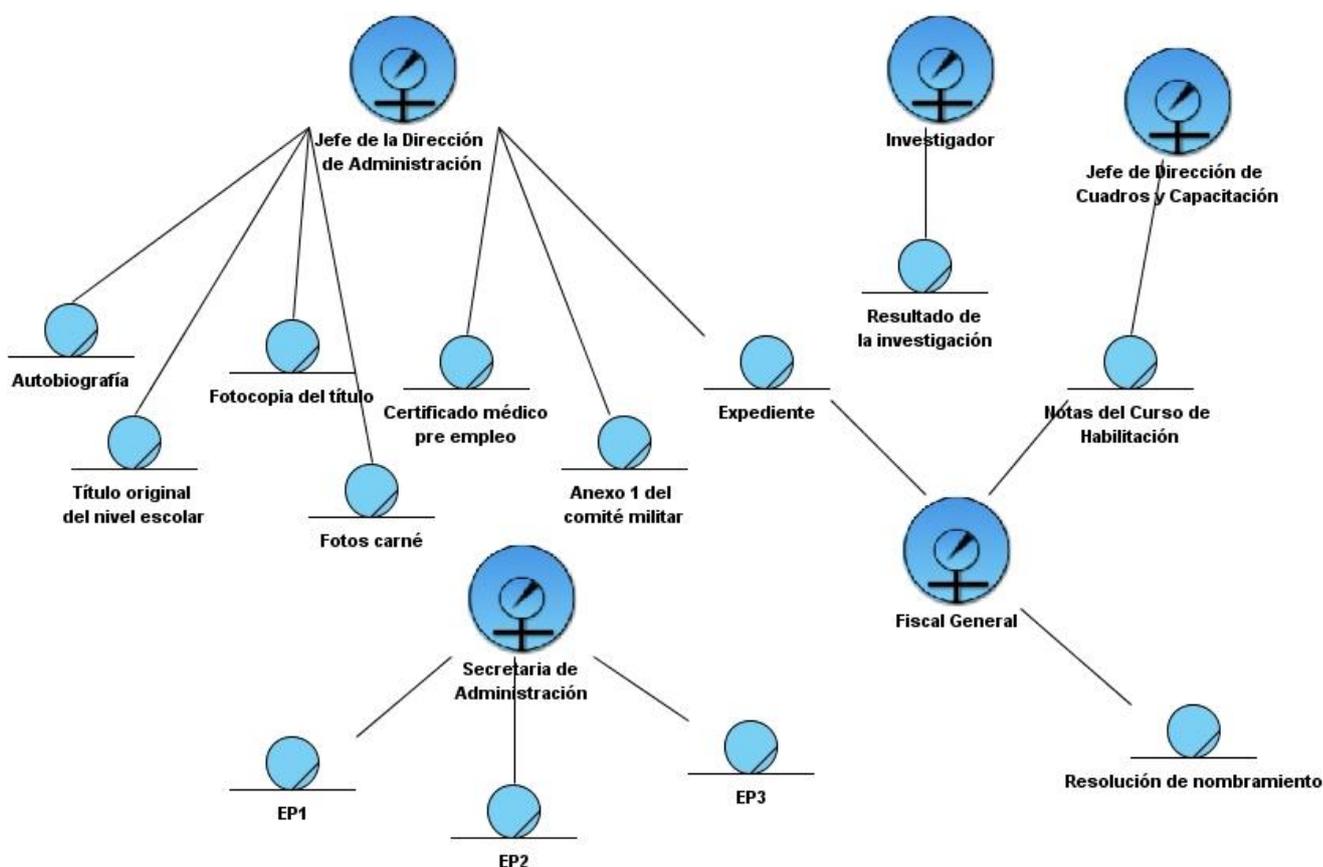
A partir de esta descripción se realizan los siguientes Diagramas de actividades que se muestran en el documento “Personal de apoyo – Modelo del negocio”.

2.3.7 Diagrama de Actividades

Un diagrama de actividades es un diagrama diseñado para mostrar una visión simplificada de lo que ocurre durante una operación o un proceso. Es una extensión de un diagrama de estados, pero el diagrama de actividades resalta precisamente a las actividades. A cada actividad se le representa con un rectángulo con las esquinas redondeadas, está dividido en calles y muestra como se utilizan las entidades que apoyan la realización de los procesos de negocio. Para consultar los diagramas de actividades del módulo Personal de apoyo, ver documento “Personal de apoyo – Modelo del negocio”.

2.3.8 Modelo de objetos

Representa la relación entre los trabajadores del negocio y las entidades. Son los objetos que los trabajadores manipulan, inspeccionan, toman, producen o utilizan durante la realización de los casos de uso del negocio.



2.4 Módulo Reportes y búsquedas

Actualmente en las fiscalías del país, la información es almacenada en archivos metálicos que deben ser consultados de forma manual, en caso de que se haga necesario recopilar algún dato relacionado con el personal que en ellas labora. Los datos estadísticos se contabilizan y actualizan continuamente y cada vez que se hace necesario enviar un reporte; después de analizada, consultada, sumada y/o recolectada la información que se les pide, lo envían por el correo o a través de llamadas telefónicas. Anualmente la Fiscalía General de la República, actualiza los datos de captación que le envían el resto de las fiscalías y con ellos, las cantidades de plazas vacantes, desocupadas y disponibles con que cuenta el país en cada una de las diferentes provincias y municipios, por tomar un ejemplo.

2.5 Flujo de trabajo de requisitos

Una vez terminado el flujo de modelado del negocio, se comienzan a capturar los requisitos que el sistema debe cumplir, tomando como entrada las actividades a informatizar identificadas en el negocio. En este epígrafe se presentan los requisitos funcionales y no funcionales de los módulos Personal de apoyo y Reportes y búsquedas. Se definen los actores del sistema, casos de uso, sus descripciones y prototipos de interfaz.

2.5.1 Requisitos funcionales

2.5.1.1 Requisitos funcionales definidos para el módulo Personal de apoyo

RF1. Registrar datos de captación de asistente fiscal.

RF3. Modificar datos de captación de asistente fiscal.

RF5. Buscar personal.

RF6. Visualizar datos de captación de asistente fiscal.

Los restantes requisitos funcionales se pueden encontrar en el documento “Personal de apoyo - Especificación de Requisitos (Requisitos funcionales)”.

2.5.1.2 Algunos requisitos funcionales definidos para el módulo Reportes y búsquedas

RF1. Realizar búsqueda avanzada del Cuadro.

RF2. Generar expediente de Personal de apoyo.

RF30. Mostrar el promedio de edad de los fiscales de cada uno de los municipios del país.

RF117. Mostrar cantidad de asistentes del fiscal a nivel nacional.

RF118. Mostrar cantidad de Jubilaciones por cada una de las provincias del país.

RF125. Mostrar cantidad de Fiscales Provinciales que tienen menos de un año de experiencia fiscal por cada una de las provincias del país.

RF185. Mostrar cantidad de fiscales listos para ser promovidos a nivel nacional.

Los restantes requisitos funcionales se pueden encontrar en el documento “Reportes y búsquedas - Especificación de Requisitos (Requisitos funcionales)”.

2.5.2 Requisitos no funcionales

Entre los requisitos no funcionales definidos para los módulos Personal de apoyo y Reportes y búsquedas se encuentran:

Software

Se deben instalar en las computadoras los siguientes programas:

PC Cliente

RNF1. Sistema operativo Linux o Windows ME/XP/2000.

RNF2. Navegadores: Internet Explorer o Mozilla Firefox.

Disponibilidad

RNF24. El sistema estará disponible en días y horarios laborables.

Soporte

RNF26. Se le debe dar un mantenimiento semanal al servidor de Base de Datos.

Seguridad

RNF31. El sistema permitirá la autenticación de usuarios.

RNF32. El sistema almacenará de forma cifrada la información confidencial generada.

RNF33. El sistema realizará salvas periódicas de la información contenida en la base de datos.

Los restantes requerimientos funcionales se pueden encontrar en el documento “Especificación de Requisitos - Requisitos funcionales”.

2.5.3 Actores del sistema

Los actores del sistema son aquellos trabajadores del negocio que tienen actividades a informatizar, o algún actor del negocio que va a interactuar con el sistema.

Actor	Descripción
Responsable	<p>Registra, modifica y visualiza los datos relacionados con la captación del Asistente fiscal y de otras personas que trabajan en la fiscalía. También puede realizar una búsqueda simple de estas personas en el sistema.</p> <p>Puede ser el Fiscal General de la República, Fiscal Jefe de Dirección de Cuadro y Capacitación, Fiscal Jefe Provincial o el Fiscal Jefe de Departamento Provincial.</p>
Lector	<p>Visualiza los datos de la captación del Asistente fiscal y de otras personas registrados en el sistema. También puede realizar una búsqueda simple de estas personas en el sistema.</p>

	Puede ser el Fiscal Jefe Departamento de Capacitación, Vice Fiscal Jefe Provincial, Vice Fiscal General o el Fiscal Jefe Municipal.
Consultor	<p>Realiza búsquedas avanzadas de cuadros, personal de apoyo y de las capacitaciones. Genera currículos y expedientes de los mismos, así como reportes que abarcan las diferentes provincias y municipios del país. También imprime reportes, expedientes y currículos.</p> <p>Puede ser el Fiscal General de la República, Fiscal Jefe de Dirección de Cuadros y Capacitación, Vicefiscal General, Vicefiscal Jefe Provincial, Fiscal Jefe Departamento Provincial o el Fiscal Jefe Municipal.</p>
Fiscal Jefe de Departamento de Capacitación	Realiza búsquedas avanzadas de capacitaciones, genera currículos y puede mostrar la disponibilidad de plazas para los asistentes fiscales. Puede imprimir la disponibilidad de plazas antes mencionada y currículos.

2.5.4 Casos de uso del sistema

Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario.

En el módulo Personal de apoyo se definen los siguientes Casos de uso:

1. Gestionar captación del Asistente fiscal.
2. Gestionar captación de Otras personas.
3. Realizar búsqueda simple.
4. Visualizar captación del Asistente fiscal.
5. Visualizar captación de Otras personas.

En el módulo Reportes y búsquedas se definen los siguientes Casos de uso:

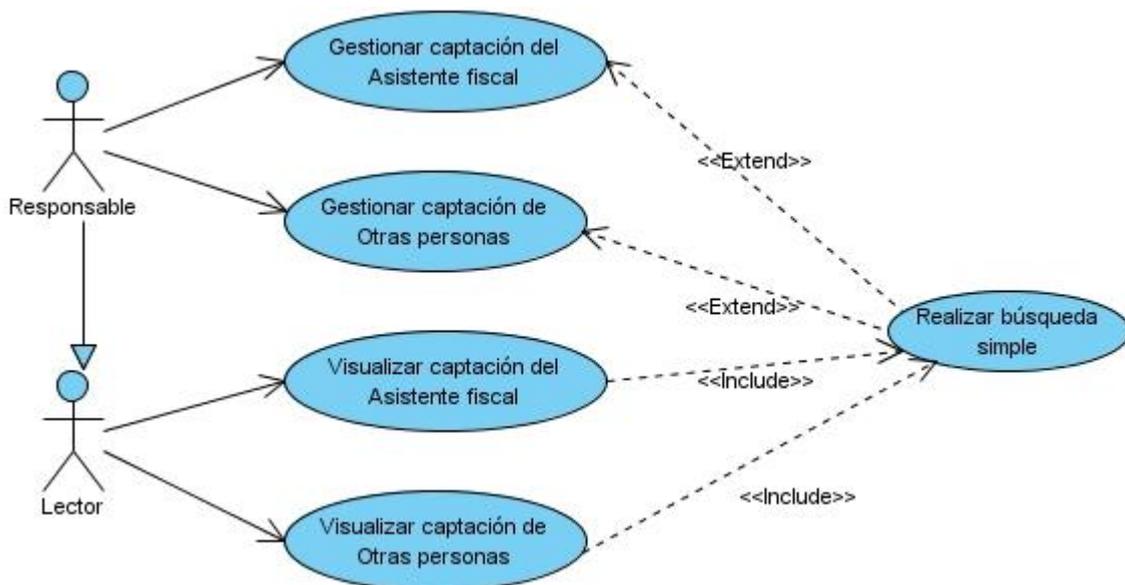
1. Realizar búsqueda avanzada.
2. Generar expediente del Personal de apoyo .

3. Generar expediente del Cuadro.
4. Mostrar fiscales por sexo y etnia.
5. Mostrar militancia de fiscales.
6. Mostrar promedio de edad de fiscales.
7. Mostrar experiencia fiscal.
8. Mostrar motivos de bajas fiscales.
9. Mostrar fiscales de la reserva.
10. Mostrar dirigentes por sexo, etnia y militancia.
11. Mostrar Asistentes fiscales y Licenciatura en derecho.
12. Realizar búsqueda avanzada de capacitación.
13. Generar currículum del Cuadro.
14. Mostrar disponibilidad de plazas para Asistentes fiscales.

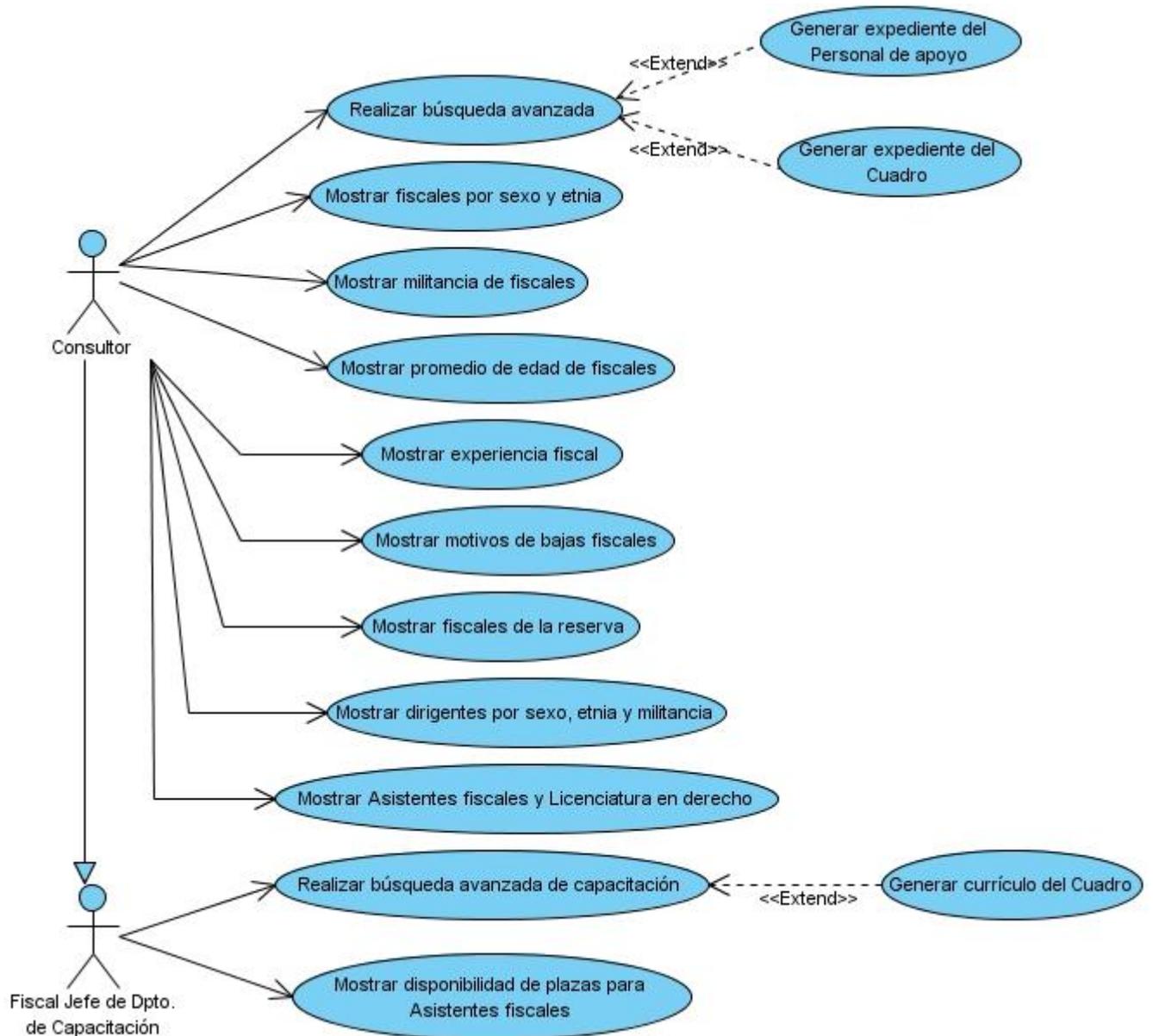
2.5.5 Diagrama de casos de uso del sistema

Los diagramas de casos de uso sirven para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas.

2.5.5.1 Diagrama de casos de uso del sistema definido para el módulo Personal de apoyo



2.5.5.2 Diagrama de casos de uso del sistema definido para el módulo Reportes y búsquedas



2.5.6 Realización de los casos de uso del sistema

A continuación se muestran tres casos de uso críticos de cada uno de los módulos: Personal de apoyo y Reportes y búsquedas con sus prototipos de interfaz, los restantes se muestran en los documentos “Modelo de Casos de Uso del sistema” de cada módulo.

2.5.6.1 Realización de los casos de uso del módulo Personal de apoyo

Descripción del Caso de Uso: Visualizar captación del Asistente fiscal

Caso de Uso:	Visualizar captación del Asistente fiscal
Actores:	Lector (inicia)
Resumen:	El proceso comienza cuando el Lector accede al sistema para leer los datos referentes a la captación del Asistente fiscal que han sido introducidos en el sistema con anterioridad. Termina cuando el sistema visualiza los mismos.
Precondiciones:	<ul style="list-style-type: none"> • El sistema debe estar correctamente instalado. • El actor debe estar autenticado con los permisos necesarios.
Referencias	RF6, RF10 CU incluido: Realizar búsqueda simple
Prioridad	Crítico

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El caso de uso comienza cuando el Lector selecciona opción Búsqueda simple de la interfaz principal.	1.1 Invoca al caso de uso Realizar búsqueda simple.
2. El Lector selecciona la opción Editar para acceder a los datos de la persona deseada.	2.1 El sistema muestra una interfaz con las opciones Modificar datos y Visualizar datos.
3. El Lector selecciona la opción Visualizar datos. Si desea cancelar ir al Flujo Alternativo al paso 3 .	3.1 El sistema muestra los datos pertenecientes al personal de apoyo seleccionado con anterioridad.
4. El Lector selecciona la opción Finalizar.	4.1 El sistema cierra la interfaz actual. 4.2 Termina el caso de uso.

Prototipo de Interfaz

Búsqueda Simple

Criterios de búsqueda:

Nombre(s)	Apellido(s)	Carné de Identidad	<input type="checkbox"/> ¿Es baja?
<input type="text"/>	<input type="text"/>	<input type="text"/>	

	Nombre(s) Apellidos Cargo	<input type="button" value="Editar"/>
	Nombre(s) Apellidos Cargo	<input type="button" value="Editar"/>

<< < 123...N > >>

Acciones de Búsqueda Simple

Acciones:

<input type="button" value="ICO"/> Modificar datos	<input type="button" value="ICO"/> Visualizar datos
---	--

Captación del Asistente del Fiscal

Datos personales:

Nombre(s) * Primer Apellido * Segundo Apellido * Edad

Carné de Identidad * Dirección particular * Teléfono particular Sexo

Provincia Municipio Correo Electrónico *

Datos laborales:

Provincia donde trabaja * Municipio donde trabaja * Ocupación *

Permanencia Esfera de trabajo Años de experiencia

Estudiando carrera universitaria: Licenciatura en derecho
 Otros

Datos de Asignaturas vencidas:

Nombre de la asignatura	Fecha de inicio	Fecha de fin
Derecho I	3/09/2004	12/01/2005

Licencia de conducción:

Número

Categoría

A: Motocicletas.
 B: Automóviles cuyo peso máximo no exceda de 3.500 kg y de 8 asientos.
 C: Automóviles para transporte de mercancías cuyo peso máximo exceda de 3.500 kg.
 D: Automóviles para transporte de personas y de 8 asientos.
 E: Vehículos cuyo tractor está comprendido en las categorías "B, C o D" que por su naturaleza no quedan incluidos en esta.

Flujo Alternativo al paso 3: "Opción Cancelar"	
Acción del Actor	Respuesta del Sistema
3a) El Lector selecciona la opción Cancelar.	3b) El sistema cierra la interfaz actual.
Poscondiciones	Se muestran los datos de captación del Asistente fiscal registrado en el sistema con anterioridad.

2.5.6.2 Realización de los casos de uso del módulo Reportes y búsquedas

Descripción del Caso de Uso: Mostrar disponibilidad de plazas para Asistentes fiscales

Caso de Uso:	Mostrar disponibilidad de plazas para Asistentes fiscales
Actores:	Fiscal Jefe de Departamento de Capacitación (inicia)
Resumen:	El proceso comienza cuando el Fiscal Jefe de Departamento de Capacitación accede al sistema y visualiza un reporte con las cantidades totales de plazas aprobadas, cubiertas y vacantes en las diferentes provincias y municipios del país para los asistentes fiscales. El Fiscal Jefe de Departamento de Capacitación imprime el reporte mostrado en caso de ser necesario.
Precondiciones:	<ul style="list-style-type: none"> • El sistema debe estar correctamente instalado. • El actor debe estar autenticado con los permisos necesarios. • Los datos de las plazas de los Asistentes fiscales están registrados en el sistema con anterioridad.
Referencias	RF187 a RF192
Prioridad	Crítico

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el Fiscal Jefe de Departamento de Capacitación selecciona el reporte Disponibilidad de plazas para Asistentes fiscales.	1.1 El sistema muestra en la interfaz correspondiente los siguientes datos: <ul style="list-style-type: none"> - Provincias (nombre de las provincias del país). - Municipios (nombre de los municipios y el Subtotal provincial del mismo). - Aprobadas (número de plazas que están aprobadas).

	<ul style="list-style-type: none"> - Cubiertas (número de plazas que están cubiertas). - Vacantes (número de plazas libres).
2. El Fiscal Jefe de Departamento de Capacitación selecciona la opción Imprimir. Si desea cancelar ir al Flujo Alternativo al paso 2.	<p>2.1 El sistema ordena imprimir el documento.</p> <p>2.2 Termina el caso de uso.</p>

Prototipo de Interfaz



Flujo Alternativo al paso 2: "Opción Cancelar"

Acción del Actor	Respuesta del Sistema
2. El Fiscal Jefe de Departamento de Capacitación selecciona la opción Cancelar.	2.1 El sistema cierra la interfaz actual.
Poscondiciones	El sistema muestra un reporte con los totales de plazas que se encuentran aprobadas, cubiertas y vacantes para cada provincia y municipios del país.

Si se desean ver las descripciones de los restantes casos de uso de los módulos Personal de apoyo y/o Reportes y búsquedas, ver el documento "Modelo de casos de uso del sistema" de ambos módulos.

2.6 Patrones de Casos de uso utilizados

A los requisitos identificados se le aplicaron los siguientes patrones de Casos de Uso: Concordancia (Reuso y Adición), CRUD (Parcial) y Múltiples actores (Roles comunes). Lo que permitió agrupar los requisitos en un menor número de casos de uso, obteniendo diagramas de casos de uso menos

engorrosos. Esto facilitó ganar en tiempo, organización y reflejar con mayor precisión las necesidades reales del cliente. A continuación se brinda una breve explicación de los patrones utilizados.

Concordancia

Extrae una subsecuencia de acciones que aparecen en diferentes lugares del flujo de casos de uso y es expresado por separado.

Adición (extend): En el caso de este patrón, la subsecuencia común de casos de uso, extiende los casos de uso compartiendo la subsecuencia de acciones. Los otros casos de uso modelan el flujo que será expandido con la subsecuencia. Este patrón es preferible usarlo cuando otros casos de uso se encuentran completos, es decir, no requieren de una subsecuencia común de acciones para modelar los usos completos del sistema.

Reuso (include): Consta de 3 casos de uso. El primero llamado subsecuencia común, modela una secuencia de acciones que aparecerán en múltiples casos de uso en el modelo. Los otros casos de uso modelan el uso del sistema que comparte la subsecuencia común de acciones. De manera que deben existir al menos dos de ellos.

CRUD (Creating, Reading, Updating, Deleting)

Este patrón se basa en la fusión de casos de uso simples para formar una unidad conceptual.

Parcial: Este patrón alternativo modela una de las vías de los casos de uso como un caso de uso separado. Es preferiblemente utilizado cuando una de las alternativas de los casos de uso es más significativa, larga o más compleja que las otras.

Múltiples actores

Roles comunes: Puede suceder que los dos actores jueguen el mismo rol sobre el CU. Este rol es representado por otro actor, heredado por los actores que comparten este rol. Es aplicable cuando, desde el punto de vista del caso de uso, solo exista una entidad externa interactuando con cada una de las instancias del caso de uso. (Övergaard & Palmkvist, 2004)

2.7 Conclusiones parciales

En este capítulo se dio cumplimiento a todas las actividades descritas en la guía propuesta para el levantamiento de requisitos. Se describieron las características de la solución propuesta partiendo de los artefactos generados en el flujo de trabajo Modelado de negocio, obteniéndose una lista de funcionalidades que debe tener el sistema, apoyada por los artefactos generados en el flujo de trabajo de Requisitos. La descripción de los casos de uso del sistema identificado, contribuyó a la obtención de una propuesta inicial de los prototipos no funcionales que posteriormente son validados con el cliente. El listado de requisitos obtenidos en el flujo de trabajo de Requisitos, dará lugar a la aplicación de técnicas de validación como las Matrices de trazabilidad, Listas de chequeo y Revisiones que permitirán demostrar que los requisitos obtenidos cubren las necesidades del cliente.

CAPÍTULO 3. VALIDACIÓN

3.1 Introducción

En este capítulo se realiza un análisis de los resultados obtenidos después de aplicar técnicas como la Validación mediante prototipos no funcionales, Revisiones, Listas de Chequeo, Matriz de trazabilidad, y Métricas, para medir la calidad de la especificación de los requisitos y la funcionalidad del diagrama de casos de uso del sistema (DCUS). Este proceso tiene como objetivo asegurar que todos los requisitos han sido establecidos sin ambigüedad, inconsistencias u omisiones, mediante la corrección de los errores detectados. De esta manera, se continuará dándole cumplimiento a la planificación del levantamiento de requisitos propuesta en el capítulo anterior.

3.2 Validación mediante Prototipos no funcionales

Los prototipos permiten validar los requisitos propuestos revelando errores u omisiones detectados en los mismos, sin tener la totalidad de la funcionalidad del sistema. Favorecen la comunicación entre clientes y desarrolladores, brindando una primera visión del producto final.

Para cada caso de uso de los módulos PA y RB se diseñaron interfaces, integradas por un grupo de pantallas que permiten verificar los requisitos que se cumplen en cada una de ellas. Se llevó a cabo una revisión minuciosa de los prototipos con la participación activa del cliente y la ayuda de la especificación de los requisitos obtenida y las descripciones de los casos de uso realizadas; se demostró que las pantallas respondían a las funcionalidades de todos los requisitos identificados.

Los prototipos pueden ser consultados en los documentos “Modelo de casos de uso del sistema” de ambos módulos.

3.3 Revisiones

El equipo de calidad del proyecto SGF, realizó la lectura y corrección de toda la documentación generada en los flujos de trabajo Modelado del negocio y Requisitos. Se señalaron errores de redacción (concordancia de ideas), falta de claridad en algunos requisitos identificados, omisiones de

tildes, casos de uso (CU) sin referencias y con errores en su estructura; según las pautas establecidas por el proyecto para la descripción de los mismos.

Todos estos errores fueron recogidos en el documento de No Conformidades (NC), el cual fue entregado al analista encargado de los módulos Personal de apoyo (PA) y Reportes y búsquedas (RB) del subsistema GCPA. Después de corregidas las NC detectadas, el equipo de calidad realizó una segunda revisión la cual concluyó con resultado satisfactorios.

3.4 Validación mediante Listas de Chequeo

Otra técnica utilizada para la validación de los requisitos fue la Lista de Chequeo. La aplicación de la misma arrojó una serie de requisitos que no habían sido descritos de forma simple, clara y/o detallada, el nombre de algunos requisitos no era el más correcto ya que no dejaban clara su funcionalidad. Además, se encontraron requisitos duplicados. Con el apoyo de las Listas de Chequeo se le dio solución a todas las no conformidades identificadas en el documento de Especificación de los requisitos de ambos módulos, obteniendo así resultados satisfactorios.

La Lista de Chequeo que se manejó se puede consultar en el documento “Lista de chequeo – Especificación de Requisitos”.

3.5 Métricas

El IEEE Standard Glossary of Software Engineering Terms define métrica como una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado. (IEEE, 1993)

Medimos para mejorar, la medición nos permite recoger la información cuantitativa que nos ayuda a identificar obstáculos, problemas de raíz, ineficiencias y otras oportunidades para mejorar la calidad del producto y el rendimiento del proceso (Pressman, 2005).

Las métricas ayudan a entender el proceso técnico que se utiliza para desarrollar un producto, éste último se mide para intentar aumentar su calidad.

3.5.1 Métricas de la calidad de la especificación

Algunos autores proponen una lista de características que pueden emplearse para valorar la calidad del modelo de análisis y la correspondiente especificación de requisitos: especificidad (ausencia de ambigüedad), complejidad, corrección, comprensión, capacidad de verificación, consistencia interna y externa, capacidad de logro, concisión, trazabilidad, capacidad de modificación, exactitud y capacidad de reutilización. Aunque estas características parecen ser cualitativas, se pueden representar usando una o más métricas.

Para determinar la especificidad de los requisitos se emplea una métrica basada en la consistencia de la interpretación de los revisores para cada requisito, para esto es necesario conocer el total de los requisitos n_r dado por:

$$n_r = n_f + n_{nf}$$

Donde n_f son los requisitos funcionales y n_{nf} los requisitos no funcionales. Luego de tener este dato se puede conocer la especificidad planteando:

$$Q = n_{ui}/n_r$$

Donde n_{ui} es el número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas, señalando que mientras más cerca esté de uno el valor de Q , menor será la ambigüedad de la especificación (Pressman, 2005).

En el presente trabajo se realizaron dos revisiones en el módulo RB y una en el módulo PA, para obtener requisitos sin incoherencias, obteniendo los siguientes resultados:

Primera revisión del módulo RB:

$$n_r = n_f + n_{nf}$$

$$n_r = 202 + 35$$

$$n_r = 237$$

Como resultado de la primera revisión se encontraron 41 requisitos que tenían problemas de redacción, que dificultaban la interpretación de los revisores y 10 redundantes que debían eliminarse. Por lo que de 237 requisitos tuvieron ambigüedad 51 de ellos.

$$Q_i = n_{ui}/n_r$$

$$Q_i = 186/237$$

$$Q_i = 0.78$$

Segunda revisión del módulo RB:

En esta revisión se solucionaron los problemas de ambigüedad eliminando los requisitos señalados como redundantes. Después de los cambios hechos se cuenta con iguales interpretaciones por parte de los revisores para los 227 requisitos que quedaron después de corregir los problemas de la primera revisión.

$$Q_i = n_{ui}/n_r$$

$$Q_i = 227/227$$

$$Q_i = 1$$

Primera revisión del módulo PA:

$$n_r = n_f + n_{nf}$$

$$n_r = 10 + 35$$

$$n_r = 45$$

En la primera revisión no se encontraron problemas en los requisitos ya que los revisores tuvieron la misma interpretación para cada uno de ellos.

$$Q_i = n_{ui}/n_r$$

$$Q_i = 45/45$$

$$Q_i = 1$$

A continuación se representan gráficamente los resultados descritos anteriormente:



Después de aplicada la métrica para la calidad de la especificación se concluye que los requisitos del módulo RB presentaban cierto nivel de ambigüedad que pudo ser eliminado, ya que en la segunda revisión realizada se comprobó que todos los problemas detectados en la primera fueron solucionados. Por este motivo se puede decir que los requisitos de los módulos PA y RB no son ambiguos.

3.5.3 Métricas para el Diagrama de Casos de Uso

Para medir la calidad de la funcionalidad del diagrama de casos de uso del sistema, se utilizó un modelo de métricas orientado a objetos. Éste consta de cuatro atributos: Consistencia, Correctitud, Completitud y Complejidad, donde:

Completitud: Grado en que se ha logrado detallar todos los casos de uso relevantes.

Consistencia: Grado en que los casos de uso del sistema describen las interacciones adecuadas entre el usuario y el sistema.

Correctitud: Grado en que las interacciones actor / sistema soportan adecuadamente el proceso del negocio.

Complejidad: Grado de claridad en la presentación de los elementos que describen el contexto y la claridad del sistema.

Cada uno de los factores tiene asociado una métrica que permite comprobar y revisar el artefacto que se esté analizando. A continuación se muestra el grupo de métricas aplicadas al diagrama de caso de uso del sistema, según las pautas para la especificación de requisitos definidas por el proyecto:

Factores de Completitud	Métricas Asociadas	Valor PA (%)	Valor RB (%)
Factor 1. ¿Se presenta una descripción resumida (descripción de alto nivel) de todos los casos de uso del negocio?	Métrica 1: Número de casos de uso que no tiene descripción resumida. [Umbral: < 10%]	0	0
Factor 2. ¿Están definidos todos los requisitos que justifican la funcionalidad del caso de uso?	Métrica 2: Número de casos de uso que tienen requisitos omitidos. [Umbral < 10%]	0	0
Factor 3. ¿Se presenta una descripción detallada (descripción extendida esencial) de todos los casos de uso del negocio?	Métrica 3: Número de casos de uso que no poseen una descripción extendida. [Umbral < 20%]	0	0
Factor 4. ¿Están todas las acciones del flujo de eventos redactadas en función del responsable?	Métrica 4: Número de casos de uso que tienen acciones del flujo de eventos no redactados en función del responsable. [Umbral < 10%]	40	35.7
Factor 5. ¿Se describen las condiciones de excepción relevantes que debe contemplar cada flujo de eventos?	Métrica 5: Número de casos de uso que no describen condiciones de excepciones relevantes. [Umbral < 20%]	0	0
Factor 6. ¿Todos los casos de uso han sido clasificados de acuerdo a su relevancia (crítico, secundario, auxiliar, opcional)?	Métrica 6: Número de casos de uso que no han sido clasificados. [Umbral: < 10%]	0	14.3
Factores de Consistencia	Métricas Asociadas	Valor PA	Valor RB

		(%)	(%)
Factor 7. ¿El nombre dado a los casos de uso es una expresión verbal que describe alguna funcionalidad relevante en el contexto del usuario?	Métrica 7: Número de casos de uso que tienen un nombre incorrecto. [Umbral < 20%]	0	14.3
Factor 8. ¿Existen acciones en el flujo de eventos asignadas a un responsable que no le corresponde?	Métrica 8: Número de casos de uso que tienen acciones del flujo de eventos asignados a un responsable que no le corresponde. [Umbral < 10%]	0	0
Factor 9. ¿Está adecuadamente redactado (en el lenguaje del usuario) el flujo de eventos?	Métrica 9: Número de casos de uso no aceptados. [Umbral < 20%]	20	0
Factor 10. ¿La descripción del flujo de eventos se inicia con la descripción de una acción externa originada por un actor o por una condición interna del sistema claramente identificable?	Métrica 10: Número de casos de uso cuya descripción extendida no inicia con una acción externa o con una condición monitoreada por el sistema. [Umbral: < 10%]	0	0
Factor 11. Si en el caso de uso interviene más de un actor, ¿existe claridad en cuál de ellos es el actor iniciador?	Métrica 11: Número de casos de uso con más de un actor que no describe cuál es el actor iniciador. [Umbral: < 20%]	0	0
Factor 12. ¿Existe una adecuada separación entre el flujo básico de eventos y los flujos alternos y/o flujos subordinados?	Métrica 12: Número de casos de uso complejos que no tienen separación del flujo básico y de flujos alternos. [Umbral: < 20%]	0	0
Factores de Correctitud	Métricas Asociadas	Valor PA (%)	Valor RB (%)
Factor 13. ¿Representa el caso de uso requisitos comprensibles por el	Métrica 13: Número de casos de uso en que los requisitos representados	0	0

usuario?	no son comprensibles por el usuario. [Umbral: < 5%]		
Factor 14. ¿Las interacciones definidas describen la funcionalidad requerida del sistema?	Métrica 14: Número de casos de uso que deben ser modificados para adecuarlos a la funcionalidad del sistema. [Umbral: < 10%]	0	0
Factores de Complejidad	Métricas Asociadas	Valor PA (%)	Valor RB (%)
Factor 15. ¿Los elementos dentro del diagrama están adecuadamente ubicados de manera que facilitan su interpretación?	Métrica 15: Número de elementos del diagrama que requieren reubicación. [Umbral: < 30%]	0	0

Estas métricas fueron aplicadas en los módulos PA y RB, arrojándose los siguientes resultados en las dos revisiones llevadas a cabo:

Primera revisión del módulo PA:

- ✓ Se detectaron 2 casos de uso que contenían acciones que no estaban redactadas en función del responsable, por lo que el atributo Completitud alcanzó el 93.3%.
- ✓ Se identificó 1 caso de uso que debía ser redactado de una manera más comprensible para los usuarios, alcanzando el atributo Consistencia el 96.7%.
- ✓ En todos los casos de uso se describieron las interacciones necesarias actor / sistema para alcanzar el objetivo del mismo, por lo que el atributo Correctitud se cumplió al 100%.
- ✓ El atributo Complejidad se cumplió al 100% ya que se comprobó que la ubicación que tienen los elementos del diagrama facilita la completa comprensión del mismo.

Como resultado de esta revisión se concluyó que a pesar de los resultados obtenidos, las métricas 4 y 9 no alcanzaron umbrales aceptables; por lo que es necesario corregir los problemas e ir a una segunda revisión para evitar que no conserven los errores detectados o surjan algunos nuevos.

Primera revisión del módulo RB:

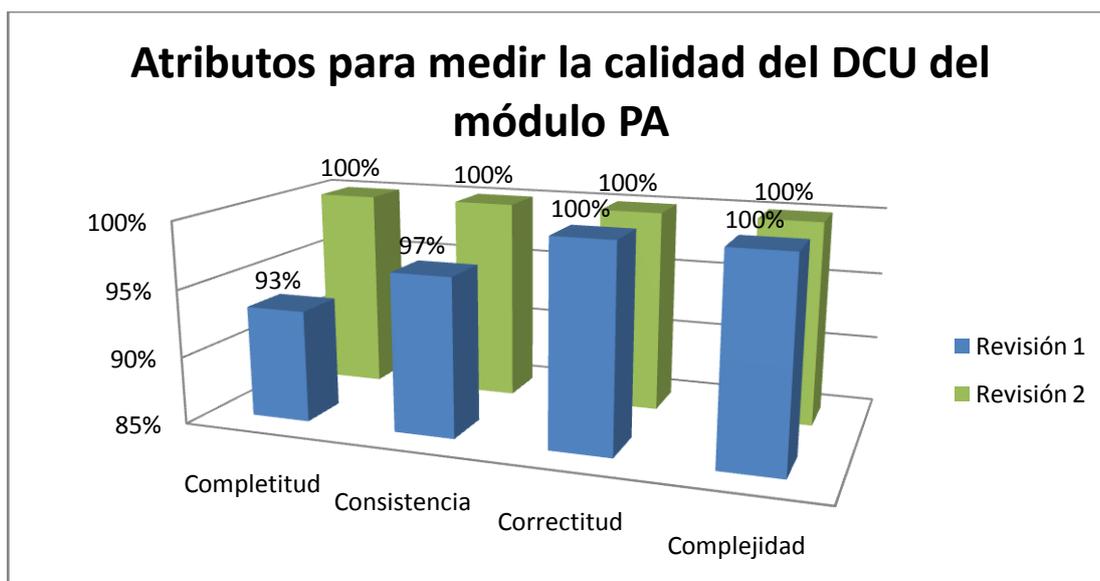
- ✓ Se detectaron 5 casos de uso que contenían acciones que no estaban redactadas en función del responsable y 2 casos de uso que no estaban clasificados de acuerdo a su relevancia, por lo que el atributo Completitud alcanzó el 91.7%.
- ✓ Se identificaron 2 casos de uso con nombres que no reflejaban claramente su funcionalidad, alcanzando el atributo Consistencia el 97.6%.
- ✓ En todos los casos de uso se describieron las interacciones necesarias actor / sistema para alcanzar el objetivo del mismo. El atributo Correctitud se cumplió al 100%.
- ✓ El atributo Complejidad se cumplió al 100% ya que se comprobó que la ubicación que tienen los elementos del diagrama facilita la completa comprensión del mismo.

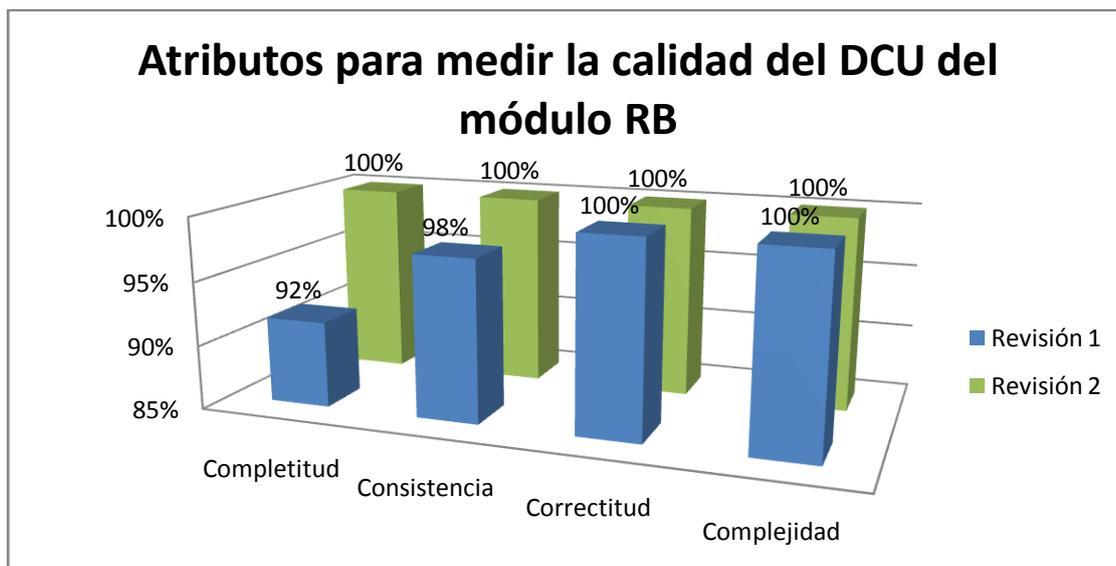
Como resultado de esta revisión se concluyó que a pesar de los resultados obtenidos, las métricas 4 y 6 no alcanzaron umbrales aceptables.

Segunda revisión de ambos módulos:

Teniendo en cuenta los resultados obtenidos en la primera revisión y después de corregidos los errores detectados en la misma, se realizó una segunda revisión en la que se comprobó que el valor 0 arrojado por todas las métricas demostró que la construcción del diagrama de casos de uso del sistema tiene el 100% de calidad.

A continuación se representan gráficamente los resultados descritos anteriormente:





3.6 Matriz de trazabilidad

El propósito de esta técnica es confirmar que cada uno de los requisitos funcionales está contenido en al menos un CU que representará su funcionalidad. Para esto se tomó como ayuda la especificación de los requisitos y el DCUS, y se utilizó la técnica conocida como Matriz de trazabilidad. Con la aplicación de la misma se verificó que todo requisito podrá ser implementado a través de algún caso de uso, y que todo caso de uso satisface algún requisito.

A continuación se muestra la matriz del módulo PA. En la misma las filas representan la lista de requisitos, mientras que las columnas representan cómo se llevan a cabo dichos requisitos en los distintos CU del diagrama.

	CU1	CU2	CU3	CU4	CU5
RF1	X				
RF2		X			
RF3	X				
RF4		X			
RF5			X		
RF6	X			X	
RF7		X			X
RF8	X				
RF9	X				
RF10	X			X	

Al aplicar esta técnica, se evidenció que todos los requisitos tienen al menos una cruz marcada, con lo que se pudo comprobar que están incluidos en algún CU. La matriz de trazabilidad demostró que los CU identificados responden a todas las necesidades del cliente.

Para ver la Matriz del módulo RB, ver documento “Reportes y búsquedas – Matriz de trazabilidad”.

3.7 Acta de liberación

Como parte del proceso de validación, los artefactos generados en los módulos Personal de apoyo y Reportes y búsquedas fueron entregados al grupo de calidad del centro CEGEL de la facultad 15 de la universidad. Se encontraron un promedio de 7 no conformidades, las cuales fueron entregadas al analista encargado de dichos módulos. Posterior a su corrección, el asesor y jefe del centro CEGEL emitió el Acta de liberación de los artefactos entregados.

3.8 Conclusiones parciales

En este capítulo después de aplicar las técnicas de validación mediante prototipos no funcionales, listas de chequeo y métricas, se identificaron errores que pudieron ser solucionados a tiempo para elevar la calidad de las etapas de elicitación, análisis y especificación, desarrolladas con anterioridad. Durante este proceso se trató de obtener los resultados más óptimos de los indicadores involucrados a través de reiteradas revisiones.

Después de validados las especificaciones de requisitos, prototipos, casos de uso y sus diagramas, se puede decir que estos cuentan con la calidad requerida para cumplir las exigencias planteadas por el cliente. Están creadas las condiciones para que el equipo de desarrollo pueda continuar en las siguientes disciplinas definidas por RUP: Análisis y Diseño e Implementación.

CONCLUSIONES GENERALES

Se realizó un estudio del estado del arte de software jurídicos que permiten el registro del personal de apoyo, realizar reportes y búsquedas relacionadas con datos curriculares y personales de los fiscales. Del mismo se concluyó que no existe actualmente en el mundo ningún software que cumpla con las características que deben cumplir los módulos Personal de apoyo y Reportes y búsquedas.

Se estudiaron además algunas metodologías existentes, lenguajes de modelado, herramientas CASE y de prototipado. Se estudió la ingeniería de requisitos, sus etapas y técnicas. A partir de este análisis y los resultados obtenidos, se trazaron distintas actividades cumplimentadas en tareas que permitieron arribar a las siguientes conclusiones:

- Llevar a cabo la etapa de elicitación ayudó a comprender los procesos del negocio de la fiscalía y a identificar algunos de los requisitos que debía cumplir el sistema.
- La aplicación de técnicas propuestas por la Ingeniería de Requisitos, contribuyó a la obtención de una lista de las necesidades del cliente. La misma apoyó el diseño mediante prototipos no funcionales de la primera versión del producto que se obtendrá en desarrollos posteriores.
- La elaboración de los artefactos: Modelo de negocio, Especificación de Requisitos del Software, Especificación de casos de uso del sistema y Prototipos de interfaz no funcional, permitieron el entendimiento común entre los clientes y desarrolladores.
- Se hizo una evaluación de la calidad de las etapas elicitación, análisis y especificación que guiaron el levantamiento de requisitos realizado. A través de las técnicas de validación y la aplicación de métricas, se elevó la calidad de la especificación de los requisitos y del DCUS obtenidos en los módulos Personal de apoyo y Reportes y búsquedas.

RECOMENDACIONES

Se recomienda a las futuras iteraciones:

- Desarrollar el Análisis, Diseño e Implementación de los módulos Personal de apoyo y Reportes y búsquedas partiendo de los artefactos obtenidos.
- Realizar la Gestión de Requisitos en los posteriores flujos de trabajo del desarrollo de los módulos Personal de apoyo y Reportes y búsquedas.

BIBLIOGRAFÍA

¿Qué es Ingeniería de Requisitos? (s.f.). Obtenido de <http://danielvn7.wordpress.com/2008/03/27/%c2%bfque-es-ingenieria-de-requisitos-ir/>

ABOGAFIN software para abogados. (s.f.). Obtenido de <http://www.softwareabogados.net/software+juridico.html>

ABOGest: Novedades. (s.f.). Obtenido de <http://www.abogest.com/>

Alonso Riverón, Y., Cruz Navarro, Y., & Tornés Medina, Y. (s.f.). *Gestiopolis*. Obtenido de <http://www.gestiopolis.com/administracion-estrategia/rup-tecnologia-aplicada-al-modelo-de-negocios.htm>

AplicacionesEmpresariales.com. (s.f.). Obtenido de <http://www.aplicacionesempresariales.com/gestiona-tus-expedientes-juridicos-con-gedex.html>

Ayuda de RUP. (s.f.).

Bernárdez, B., Durán, A., & Toro, M. (2004). Una propuesta para la verificación de requisitos basada en métricas.

Bernárdez, B., Durán, A., & Toro, M. (Agosto de 2004). *Una propuesta para la verificación de requisitos basada en métricas*. Recuperado el 11 de 05 de 2008, de Revista de Procesos y Métricas de las Tecnologías de la Información: <http://www.aemes.org/rpm/descargar.php?volumen=1&numero=2&articulo=2>

Case. (s.f.). Obtenido de <http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5.htm>

Dávila, N. D. (2003). *Ingeniería de Requerimientos: Una guía para extraer, analizar, especificar y validar los requerimientos de un proyecto*.

Durán, A., Bernárdez, B., Ruiz, A., & Toro, M. (1999). *A Requirements Elicitation Approach Based in Templates and Patterns*. Buenos Aires, Argentina.

Escalona, J. M., & Koch, N. (2002). *Ingeniería de Requisitos en Aplicaciones para la Web - Un estudio comparativo*.

Escribano, F. G. (2002). *Introducción a Extreme Programming*. Obtenido de <http://www.info-ab.uclm.es/asignaturas/42551/trabajosAnteriores/Presentacion-XP.pdf>

Gedex.net. (s.f.). Obtenido de <http://www.brindys.com/gedex/>

Gil, G. D. (2002). *Herramienta para implementar LEL y escenarios (TILS)*. Argentina.

Gómez, J. (s.f.). *Softonic*. Obtenido de <http://axure-rp.softonic.com/>

Hurtado, L. (2006). *Planeación de Proyectos - Planeación y estimación de Proyectos Informaticos*. Obtenido de http://www.wikilearning.com/articulo/planificacion_de_proyectos-planeacion_y_estimacion_de_proyectos_informaticos/9597-1

IEEE. (1993). IEEE Standars Collection: Software Engineering.

IEEE'98a. *Guide for developing System Requirements Specifications, IEEE, 1998a. 1233-1998: 36*.

IEEE'98b. *Recommended Practice for Software Requirements Specifications, IEEE, 1998b. 830-1998: 37*.

Importancia de la captura de requerimientos en el proceso productivo del proyecto CISCOP de la facultad regional de Artemisa. (s.f.). Obtenido de <http://www.articuloz.com/otro-articulos/importancia-de-la-captura-de-requerimientos-en-el-proceso-productivo-del-proyecto-ciscop-de-la-facultad-regional-de-artemisa-importance-of-the-captu-636738.html>

INEI, I. N. *Herramientas Case*.

Insfrán, E., Pastor, O., & Wieringa, R. (2002). *Requirements Engineering-Based Conceptual Modeling* (Vol. VII).

Jacobson, I. (1995). *Modeling with use cases-Formalizing use-case modelling*.

Jacobson, I., & Booch, G. (2004). *El Proceso Unificado de Desarrollo de Software* (Vol. I). Félix Varela.

Jacobson, I., Booch, G., & Rumbaugh, J. (2000). *El Proceso Unificado de Desarrollo de Software.*

La Ingeniería de Software. (s.f.). Obtenido de <http://www.angelfire.com/scifi/jzavalar/apuntes/IngSoftware.html>

Larman, C. (1999). *UML y Patrones.*

Lexnet. (s.f.). Obtenido de <http://www.anf.es/anf/certificacion/proyectos-especiales/200.1.123.html>

Loucopoulos, P., & Karakostas, V. (1995). *System Requirements Engineering.* Londres.

Lowe, D., Hall, W., & John, W. (1999). *An Engineering approach.*

Mendoza, M. A. (2004). *Metodologías De Desarrollo De Software.* Obtenido de http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html

Microsoft Office Online. (s.f.). Recuperado el 2010, de <http://office.microsoft.com/es-es/visio/HA101656403082.aspx>

Olsina, L. (1999). *Metodología cualitativa para la evaluación y comparación de la calidad de sitios web.*

Övergaard, G., & Palmkvist, K. (2004). *Use Cases: Patterns and Blueprints.*

Pan, D., Zhu, D., & Johnson, K. (2001). *Requirements Engineering Techniques.*

Pérez, J. D. Notaciones y lenguajes de procesos. Una visión global.

Piattini, M. G., Calvo-Manzano, J. A., Cervera, J., & Fernández, L. (1996). *Análisis y diseño detallado de Aplicaciones informáticas de gestión.*

Pressman, R. S. (2005). *Ingeniería de Software: Un enfoque práctico.*

Pressman, R. S. (2005). *Ingeniería del Software: Un enfoque práctico.*

Pressman, R. S. (2005). *Ingeniería del Software: Un enfoque práctico*. Mc Graw Hill.

Proceso de Desarrollo de Software. (s.f.). Obtenido de http://webcache.googleusercontent.com/search?q=cache:VDccWRFNAB0J:www.dsic.upv.es/asignaturas/facultad/lsi/doc/IntroduccionProcesoSW.doc+%22Proceso+de+desarrollo+de+software%22&cd=2&hl=es&ct=clnk&gl=cu&lr=lang_es

San Germán, M. J. (2003). Métodos de modelado ideo e ideo3 y uso básico.

Schmuller, J. (2000). *Aprendiendo UML en 24 horas*.

Sitio Oficial Fiscalía General de la República. (s.f.). Recuperado el Enero de 2009, de http://www.fgr.cu/index.php?option=com_content&view=article&id=45&Itemid=74

Vilain, P., Schwabe, D., & Sieckenius, C. (2000). *A diagrammatic Tool for Representing User Interaction in UML*.

Young, R. R. (2004). *The Requirements Engineering Handbook*. Londres .

Zaragoza, U. d. (s.f.). *Prototipado*. Obtenido de <http://www.sidar.org/recur/desdi/traduc/es/visitable/maner/Prototipado.htm>

GLOSARIO DE TÉRMINOS

Abogado: El abogado pertenece a una organización autofinanciada que representa a las personas en su actuar cotidiano ante situaciones que demandan la presencia letrada o que requieren conocimientos jurídicos.

Fiscal: Abogado que representa al gobierno (local, estatal o federal) en causas penales. Tiene como funciones la de velar por el cumplimiento de la legalidad (por personas naturales o jurídicas) y el ejercicio de la acción penal pública en representación del Estado. Rinde cuentas de su actuar a la Asamblea Nacional.

Demo: Versión demostrativa de un tipo de software. Es una versión restringida, que muestra una parte de todas las posibilidades que ofrece la aplicación.

MTSS: El Ministerio de Trabajo y Seguridad Social (MTSS), es un Organismo de la Administración Central del Estado, según dispone el Decreto Ley 147 de la reorganización de los organismos de la Administración Central del Estado, del 21 de abril de 1994, en su artículo 10. Tiene como misión Desarrollar el pleno empleo, la Gestión Estratégica de los Recursos Humanos para el logro de un alto desempeño humano, económico y social, y el nuevo modelo de gestión de la Seguridad Social.

IEEE: Corresponde a las siglas de The Institute of Electrical and Electronics Engineers (el Instituto de Ingenieros Eléctricos y Electrónicos), es la asociación técnica y profesional, sin fines de lucro, más grande del mundo formada por profesionales de todas las disciplinas de la ingeniería. Su trabajo es promover la creatividad, el desarrollo y la integración, compartir y aplicar los avances en las tecnologías de la información, electrónica y ciencias en general para beneficio de la humanidad y de los mismos profesionales.

PDF: acrónimo del inglés Portable Document Format (formato de documento portátil), es un formato de almacenamiento de documentos, de tipo compuesto (imagen vectorial, mapa de bits y texto). Está especialmente ideado para documentos susceptibles de ser impresos, ya que especifica toda la información necesaria para la presentación final del documento, determinando todos los detalles de cómo va a quedar, no requiriéndose procesos anteriores de ajuste ni de maquetación.

XML: Acrónimo de Extensible Markup Language («lenguaje de marcas ampliable»), es un metalenguaje extensible de etiquetas, aunque no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

XPS: Acrónimo de XML Paper Specification. Es un formato de archivo electrónico que conserva el formato del documento y permite compartir archivos. El formato XPS garantiza que, cuando el archivo se vea en línea o impreso, se conservará exactamente el formato original y que los datos del archivo no se puedan copiar o cambiar con facilidad.

Release: Producto final preparado para lanzarse como versión definitiva a menos que aparezcan errores que lo impidan. Implementa todas las funciones del diseño y se encuentra libre de cualquier error que suponga un punto muerto en el desarrollo.

Artefacto: Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.

Cliente: Aquella persona o empresa que contrata al desarrollador de software.

No Conformidad: Según la definición de ISO 9000: 2000 (3.6.2), una no conformidad es el “incumplimiento de un requisito”.

Actor: un actor es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio y el sistema interactúan.

Caso de Uso: Representa a un proceso, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores. Desde la perspectiva de un actor individual, define un flujo de trabajo completo que produce resultados deseables.

ANEXOS

Anexo 1. Comparación entre las metodologías ágiles y las metodologías tradicionales.

Metodologías ágiles	Metodologías tradicionales
Especialmente preparados para cambios durante el Proyecto.	Cierta resistencia a los cambios.
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos.	Más artefactos.
Pocos roles.	Más roles.
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos.