

# Universidad de las Ciencias Informáticas



Facultad 3

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

**Título:** Análisis y diseño del proceso Quejas y Reclamaciones  
para el Sistema de Informatización de la Gestión de las Fiscalías II

**Autor(es):**

Yamilé Pérez Paz

Yusmila Cruz Fernández

**Tutor:** Ing. Yenier Figueroa Machado

**Co-tutor:** Ing. Eduardo Castillo Benítez

La Habana, Junio de 2011



*“Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber.”*

*Albert Einstein.*



# *Declaración de Autoría*

---

Declaramos ser los únicos autores de este trabajo y autorizamos a la Facultad 3 de Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

## **Autores:**

Yusmila Cruz Fernández

Yamilé Pérez Paz

\_\_\_\_\_

\_\_\_\_\_

**Firma del Autor**

**Firma del Autor**

## **Tutor:**

Ing. Yenier Figueroa Machado

\_\_\_\_\_

**Firma del Tutor**



## *Datos de contacto*

---

**Datos de tutor:**

**Nombre:** Ing. Yenier Figueroa Machado

**Correo electrónico:** *yfigueroa@uci.cu*

**Datos del co-tutor:**

**Nombre:** Ing. Eduardo Castillo Benítez

**Correo electrónico:** *ecbenitez@uci.cu*

**Datos del autor:**

**Nombre:** Yamilé Pérez Paz

**Correo electrónico:** *yppaz@estudiantes.uci.cu*

**Datos del autor:**

**Nombre:** Yusmila Cruz Fernández

**Correo electrónico:** *ycfernandez@estudiantes.uci.cu*



## Agradecimientos

---

*Uf... Créanme que ésta es la parte más difícil de todas. Tengo en mi corazón un manojito de sentimientos que de plasmarlos en este papel no tendría para cuando terminar pues hay personas en mi vida a las que no tendría con qué pagarle lo que han hecho por mí... Aquí van:*

*A: Mi papito lindo por ser mi ejemplo a seguir, porque sin su apoyo incondicional no hubiese podido llegar hasta aquí, por darme lo que he necesitado, por sus consejos, por consentirme y malcriarme tanto. Por encaminarme en la vida y ayudarme a ser una mujer. Creo que si volviera a nacer y tuviera que elegir un padre, te volvería a elegir... Te amo y siempre estarás en mi corazón.*

*A mi mami de crianza, mami Ramona, porque supo criarme, darme el amor, el cariño, la ternura y todas esas cosas que lejos de ser mi abuela me hacen sentir como su hijita más pequeña. Gracias mamita por estar, por todo lo que has hecho por mí, por consentirme, por tenerme confianza y estar siempre pendiente de mí. Sabes que te amo mucho y que tienes un lugar especial en mi corazón.*

*A mi mamá porque a pesar de nuestras diferencias supo aceptarme y estar a mi lado en los momentos más difíciles. Porque cuando necesité revelar un secreto fuerte estuvo ahí para apoyarme y darme un consejo. Porque después de tanto supo darme tranquilidad y confianza para seguir adelante. Gracias mami. Te quiero.*

*A mi hermanito Pedro con su cariño ha sabido insertarse en mis seres más queridos, eres mi único hermanito y gracias a tí he podido experimentar este sentimiento tan lindo. Gracias por existir. Te quiero mucho.*

*A mis tíos, mi tío el bolo como todos cariñosamente le llamamos porque a tu lado crecí y supiste quererme como tu hermanita más pequeña. Gracias por estar ahí, sé que puedo contar contigo. Te quiero mucho. A mi tío Tony porque siempre me buscó el libro que necesité para estudiar las matemáticas, también te quiero mi vaquita pinta jejeje.*

*A Yailén y Mirelis que han sabido ganarse el cariño y admiración de toda la familia, la quiero.*

*A mi padrastro Alexander por cuidar de mi mamá y mi hermanito, por su apoyo y su aptitud en los momentos más difíciles.*

*A mi abuela por parte de padre, mi tía Yairis que siempre me ha aconsejado mucho y aunque hoy no esté aquí, ella sabe que la quiero y la llevo en mi corazón.*

*A mis tías Caruca, Dora, Aleida, Adita, Dania y María por el cariño y la confianza.*

*A mis hermanitas de infancia Geisy e Ismaray. Las quiero mucho... Ismy: gracias por estar ahí cuando pero me he sentido, por hacerme sentir que tengo con quien contar, porque me has dado el cariño de la hermana hembra que no tengo. A mi*



## Agradecimientos

---

*segunda familia, tu mami Eva, Carlito, Celina, Israel, son mis vecinos del alma que me vieron crecer y con tanto cariño me han acogido en esa familia... también los quiero mucho...*

*A Moya por su apoyo y confianza, por ser parte de mi formación, por quererme y apoyarme tanto...*

*A mis primos, en especial a mi primo Julito por sus consejos, el cariño y su amor incondicional.*

*A mis compañeros de estudio y amigas de convivencia: Sandrita, Dailín, Yunet, que aunque no se encuentre aquí la queremos y la recordamos, Mailín, Yamilé, Ari, gracias por aguantarme y ayudarme a salir adelante.*

*Al profe Yudel fue un impulso para mí en mi tercer año de la carrera cuando todo mi mundo era oscuro y no supe qué hacer. Estuvo ahí para darme fuerzas. Gracias Yude. Te aprecio.*

*A papi, mi tutor, por el carisma y la confianza de que las cosas salieran siempre bien, a las profes Anna y Yanet por su apoyo.*

*A mi novio porque en muy poco tiempo ha logrado robarse un pedacito de mi corazón, ha sabido aguantar mis malcriadeces y darme el cariño que necesito para seguir, por desestresarme en las noches con los mejores de sus trucos de magia ejejej. Por hacerme ver la vida de otra forma y hacer de mis 24 hrs diarias, momentos felices e inolvidables. Te quiero mi tito.*

*A mi compañera de tesis por su paciencia y apoyo más lo necesité, por aguantarme mis malos genios en los momentos más estresantes y porque después de tanto tuve que aceptarla como amiga ejejej. Gracias mi flaquí, sin ti no hubiese sido posible...*

*No por último son menos importantes mis agradecimientos a la persona que más he amado en mi vida, la persona que supo darme amor, cariño, confianza, educación, respeto, que me supo dar mucho en lo poco... Que tantas mañanas me despertó con un maruco para ir a la escuela, por criarme para que yo fuera hoy la persona que soy, por aquellos abrazos que hoy no están, por su amor de padre enamorado de mis éxitos... Mi papito lindo, mi papi de oro, sé que te hubiese encantado verme hoy así, verme graduada como siempre lo soñaste, como me dijiste en la cama unos días antes de morir.....Te amo.*

**Yami**



# Agradecimientos

---

*A mi mamá y a mi papá por su entrega y dedicación, por ayudarme en todo y siempre apoyarme en todas las decisiones, por ser unos excelentes padres, los quiero mucho.*

*A mi hermano que siempre esta ahí cuando lo necesito y por aguantarme siempre mis malcriadeces.*

*A toda mi familia que siempre me ha querido tanto y se sienten muy orgullosos de mí, a mis primos que espero les sirva de ejemplo para lograr lo que quieran.*

*A mis abuelos y tíos en especial mi tío Rogelio que ha sido para mí también un padre y yo una hija para él siempre complaciéndome en todo.*

*A mi tía Esther, mi abuela Francisca que me quiere con la vida al igual que yo a ella, a una persona que he querido mucho y ya no esta presente, mi abuelo Tomas, siempre me cuidó, educó y malcrió desde chiquita.*

*Agradecer a mi cuñada, a todos mis vecinos que siempre estuvieron muy preocupados por mí y me quieren mucho, especialmente al Barce por su cariño incondicional.*

*A mis amigas y hermanas Yunet, Yudelís, Mailín, Dailín y Semidaris que siempre han estado conmigo en los buenos y malos momentos siempre preocupadas por mí y dándome su apoyo en todo, por soportarme durante todos estos años.*

*A mis otras hermanitas de la infancia Dayanay, Graydis y Yaneisy.*

*A todos mis compañeros con los que compartí desde un aula hasta las mejores fiestas.*

*A mis tutores por su incondicional ayuda y mis compañeros de proyecto y a todos los profesores que de una forma u otra aportaron un granito de arena en mi formación.*

*A mi equipo del voli con el cual compartí momentos emocionantes, Yamile que siempre estaba arriba de mí para ir a los entrenamientos, al profe y amigo Argilagos que siempre me tuvo como su alumna preferida y no por ser buena y mucho menos la mejor jugando.*

*A los fiscales por brindarnos su atención y ayudarnos a que el trabajo saliera adelante.*

*Agradecer muy especial a mi compañera que sin ella no lo hubiera logrado, siempre muy preocupada y ayudándome en todo, mejor no la hubiera querido.*

*Yusmila*



## *Dedicatoria*

---

*Dedico esta tesis a mi papi lindo, A mi mami de crianza Ramona, A mi mamá, a mi hermanito.*

*A la memoria de mi papi Pedro.*

*A mis padres que siempre confiaron en mí.  
A mi tío Rojo que hoy esta viendo una hija graduarse. A toda mi familia, mis tíos, primos y abuelos.*

*Yami*

*Yusmila*





Como resultado de los esfuerzos realizados a lo largo del proceso revolucionario cubano se han logrado avances en la informática, siendo uno de ellos el Sistema de Informatización para la Gestión de Fiscalías, el cual tiene como objetivo informatizar los procesos que tienen lugar en la Fiscalía General de la República (FGR).

El objetivo del trabajo es realizar el análisis y diseño de una solución informática para el Proceso Quejas o Reclamaciones en las materias Penal, Civil y de Familia, Administrativas y Laborales en los que interviene el fiscal, el cual pertenece al área de Protección de los Derechos Ciudadanos de la FGR.

Se realiza este trabajo teniendo como guía el ciclo de vida para proyectos propuesto por el Programa de Mejora que se lleva a cabo en la universidad, desarrollando específicamente las disciplinas Modelado del Negocio, Requisitos y Análisis y Diseño. Se obtuvo como resultado la realización y validación del análisis y diseño del sistema generándose los artefactos correspondientes.

El presente trabajo de diploma consta de cuatro capítulos. En el primero se puntualizan asuntos referentes a la FGR y a elementos del proceso de desarrollo de software. En el capítulo dos, se muestra lo referente al Modelado del Negocio en cuanto a artefactos generados y su validación. En el tercero, se muestran los resultados obtenidos en la disciplina de Requisitos y la validación de los mismos y por último, en el capítulo cuatro, se presentan los artefactos generados durante el Diseño y la validación de los mismos.

## **PALABRAS CLAVES**

Fiscalía General de la República (FGR), Quejas o Reclamaciones, Programa de Mejora, Modelado de Negocio, Requisitos, Diseño.



## ÍNDICE DE CONTENIDOS

INTRODUCCIÓN.....	1
<b>CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA .....</b>	<b>1</b>
1.1    Introducción.....	1
1.2    Conceptos fundamentales .....	1
1.3    Tendencias internacionales de las quejas o reclamaciones .....	3
1.4    Software existentes que informatizan el proceso Quejas o Reclamaciones .....	4
1.5    Proceso de Desarrollo de Software .....	5
1.5.1    Aspectos importantes de CMMI .....	6
1.5.2    Ciclo de vida del PM .....	8
1.6    Conceptualización de las técnicas utilizadas .....	11
1.7    Definiciones teóricas de los patrones utilizados.....	12
1.8    Métricas .....	17
1.9    Herramientas propuestas por el Programa de Mejora .....	20
1.10    Notación y lenguaje de modelado .....	21
1.11    Arquitectura del sistema .....	22
1.12    Conclusiones parciales.....	23
<b>CAPÍTULO II. MODELADO DEL NEGOCIO .....</b>	<b>24</b>
2.1    Introducción.....	24
2.2    Modelado del Negocio.....	24
2.2.1    Resultados de la aplicación de las técnicas .....	26
2.2.2    Patrones de negocio .....	26
2.2.3    Descripción de subprocesos de negocio.....	28
2.2.4    Modelo Conceptual .....	31
2.2.5    Validación del modelado de negocio del proceso Quejas o Reclamaciones .....	31
2.3    Conclusiones parciales.....	32
<b>CAPÍTULO 3. ESPECIFICACIÓN DE REQUISITOS .....</b>	<b>33</b>
3.1    Introducción.....	33
3.2    Fases de la disciplina .....	33
3.2.1    Obtención de requisitos .....	33
3.2.2    Análisis de requisitos.....	39
3.2.3    Especificación de requisitos .....	39



# Tabla de contenidos

---

---

3.2.4	Validación de requisitos del proceso Quejas o Reclamaciones .....	45
3.2.5	Administración de requisitos .....	50
3.3	Conclusiones parciales .....	50
<b>CAPÍTULO IV. DISEÑO .....</b>		<b>51</b>
4.1	Introducción.....	51
4.2	Diseño .....	51
4.2.1	Diagrama de paquetes del diseño .....	53
4.2.2	Diagrama de clases del diseño .....	53
4.2.3	Validación del diseño del proceso Quejas o Reclamaciones .....	56
4.3	Conclusiones Parciales .....	59
<b>CONCLUSIONES GENERALES .....</b>		<b>60</b>
<b>RECOMENDACIONES .....</b>		<b>61</b>
<b>BIBLIOGRAFÍA .....</b>		<b>62</b>
<b>ANEXOS .....</b>		<b>¡ERROR! MARCADOR NO DEFINIDO.</b>
<b>GLOSARIO DE TÉRMINOS .....</b>		<b>65</b>

## ÍNDICE DE FIGURAS

Figura Nº 1 Diagrama de Procesos de Negocio de las Quejas o Reclamaciones General .....	25
Figura Nº 2 Patrón Secuencia .....	27
Figura Nº 3 Patrón Distribución en Paralelo .....	27
Figura Nº 4 Patrón Selección Exclusiva .....	27
Figura Nº 5 Patrón Mezcla Múltiple o Serialización .....	28
Figura Nº 6 Diagrama de Procesos de Negocio del subproceso Asuntos administrativos y laborales. ....	28
Figura Nº 7 Diagrama de Casos de Uso del Sistema. ....	42
Figura Nº 8 Patrón Roles Comunes .....	43
Figura Nº 9 Diagrama de Paquetes .....	53
Figura Nº 10 Diagrama de Clases del Diseño. ....	55

## ÍNDICE DE TABLAS

Tabla 1 Áreas de proceso de CMMI. Nivel 2 .....	7
Tabla 2 Clasificación de las Clases .....	19
Tabla 3 Descripción de procesos de negocio del sub-proceso Asuntos administrativos y laborales. ....	29
Tabla 4 Especificación de Requisitos para el requisito Crear Acta de Comparecencia .....	40
Tabla 5 Especificación de casos de uso para el CU Registrar Queja o Reclamación. ....	43
Tabla 6 Resultados de la métrica para evaluar los casos de uso .....	47
Tabla 7 Estereotipos web .....	54
Tabla 8 Descripción de las clases del diseño. ....	56
Tabla 9 Representación del tamaño de las clases del diseño. ....	57
Tabla 10 Resultados de la Métrica Tamaño de Clases .....	58

## INTRODUCCIÓN

En medio de un grupo de transformaciones educacionales y sociales se ha llevado a cabo un profundo proceso de mejoras elevando el nivel científico cultural, así como fomentar las raíces educativas y cognoscitivas. En este proceso se encuentra en primera línea la Universidad de las Ciencias Informáticas (UCI) donde se desarrollan diferentes actividades académicas y productivas teniendo alta responsabilidad en la preparación integral de profesionales de la informática investigando, produciendo software y servicios informáticos para la sociedad cubana y el mundo.

La producción de software y la tarea de informatizar las diferentes áreas del país, recae principalmente en los proyectos productivos de la UCI, siendo uno de estos el SIGEF, creado con el objetivo de realizar una aplicación que optimice la calidad de tramitación, supervisión y control en tiempo de los procesos judiciales cubanos teniendo como alcance obtener una fuerza fiscal óptima y eficiente con mayor economía y seguridad en cada uno de sus procesos.

La FGR es una de las instituciones encargadas de atender las quejas o reclamaciones que en el orden legal formulan los ciudadanos sobre las presuntas violaciones de sus derechos. El Reglamento de la Ley de la FGR, Instrucción No 10/99, establece que el Dpto. de Protección de los Derechos Ciudadanos (PDC) tiene entre sus funciones la dirección metodológica y el control del trabajo que realizan los órganos de la fiscalía en lo que se refiere a la tramitación, investigación y respuesta que en el orden legal proceda a las quejas y reclamaciones de la población.

Existe una primera fase del proyecto (SIGEF I) en la que se informatizaron algunos de los procesos que se llevan a cabo en la FGR, dentro de los cuales no se encuentran los procesos que abarcan las quejas o reclamaciones. Las quejas se tramitan según la materia a la que corresponda. Existen varias materias por las que se puede tratar la queja, estas son: Penal, Civil y Familia, Administrativa y Laboral haciéndose corresponder cada una con un proceso. Con el objetivo de obtener resultados tangibles a corto plazo, se incluye el Proceso Quejas y Reclamaciones en una segunda fase del proyecto lo que significará una reducción del tiempo para la realización de las diligencias fiscales en estos procesos resultando amena la búsqueda de documentos en formato digital y realizarlo de una forma más eficiente en su momento. Para llevar a cabo estas operaciones se necesita de una cantidad considerable de fiscales, de los que actualmente no dispone el país. Debido a esto, se hace engorroso el trabajo cuando la realización y el control de todos los procesos requieren más tiempo del esperado ya que diariamente se genera un gran

número de documentos, viéndose afectado el análisis de los mismos de manera rápida y efectiva. Por su parte, toda la documentación que requiere la queja se encuentra manuscrita por lo que al paso del tiempo se deteriora, mostrando el contenido en tonos borrosos. Se deduce así que el problema más grande y la importancia de lograr informatizar estos procesos, se enfoca en que toda la información está manuscrita, haciéndose difícil la búsqueda inmediata de una documentación determinada. Pasado los cinco años de permanencia de estos documentos en la fiscalía se procede a la destrucción de los mismos, imposibilitando su acceso en tiempos futuros. Son estos los factores que hacen real la necesidad por parte de la FGR de informatizar los procesos fiscales, en este caso, las quejas o reclamaciones de la población.

Actualmente en la FGR se trabaja con un sistema de quejas que no satisface las necesidades de trabajo de los fiscales en cuanto a trámite de la reclamación. Éste sistema solamente da la posibilidad de hacer registros y búsquedas en determinadas materias que finalmente no logran abarcar todo el círculo de trabajo de los fiscales. En lo adelante se detallan las funcionalidades de esta aplicación.

Por todo lo antes planteado se necesita crear un mecanismo que facilite el registro de esta información, agilice y haga más efectivo el trabajo de los fiscales a lo largo de todo el territorio. Para ello es menester agregarle nuevas funcionalidades al sistema ya existente (SIGEF I), de manera que se logre dar una solución eficiente a los problemas anteriormente tratados. Teniendo en cuenta que el sistema ya implementado se rige por una arquitectura definida en el proyecto, se preverá que las nuevas funcionalidades logren insertarse perfectamente en el mismo, obteniendo como resultado mejores servicios y funcionalidades haciendo uso de las Tecnologías de la Información y las Comunicaciones (TIC) que juegan un papel protagónico en este campo para la informatización y automatización de la mayoría de las esferas y procesos de la producción y los servicios.

De todo lo anterior se identifica como **problema de la investigación** el siguiente: ¿Cómo llevar el proceso Quejas y Reclamaciones a elementos que faciliten su implementación para el Sistema de Informatización de la Gestión de las Fiscalías?

Luego de un análisis del problema antes planteado se tiene como **Objeto de estudio** el Proceso de desarrollo de software. Definiendo como **Objetivo general** el siguiente: Realizar el análisis y diseño que permita la posterior implementación del proceso Quejas y Reclamaciones para el Sistema de Informatización de la Gestión de las Fiscalías. Teniendo como **Campo de acción**: Análisis y diseño del proceso Quejas y Reclamaciones. Deduciéndose entonces como **Idea a Defender** la siguiente:



# Introducción

---

Realizando el Análisis y Diseño del proceso Quejas y Reclamaciones se logrará llevar el mismo a elementos que faciliten su implementación.

Para darle cumplimiento al objetivo general, se trazaron los siguientes **Objetivos Específicos**:

1. Elaborar el marco teórico de la investigación de manera que permita enfocar la misma a los términos que serán utilizados.
2. Desarrollar y validar el modelado del negocio permitiendo obtener los artefactos con la calidad requerida.
3. Desarrollar y validar los requisitos de forma que se obtengan los artefactos con la calidad requerida.
4. Desarrollar y validar el diseño de forma que facilite obtener los artefactos con la calidad requerida y sean útiles para la implementación.

Para cumplir los objetivos específicos antes expuestos se realizarán a lo largo de todo el proceso investigativo las siguientes **tareas**:

1. Estudio de los procesos referentes a Quejas o Reclamaciones que tienen lugar en la FGR.
2. Estudio del Programa de Mejora para llevar a cabo el proceso de desarrollo de software, los lenguajes de modelado y herramientas que propone el mismo para generar los artefactos correspondientes.
3. Identificación de los procesos de negocio.
4. Modelación de los procesos de negocio.
5. Descripción de los procesos de negocio.
6. Realización del modelo conceptual.
7. Elaboración del mapa de procesos.
8. Identificación de los Requisitos de Software.
9. Especificación de los requisitos de software.
10. Descripción de los Requisitos de Software.



# Introducción

---

11. Refinamiento del Modelo Conceptual.
12. Realización de los prototipos de interfaz de usuario para validar los requisitos identificados.
13. Elaboración del Diagrama de Casos de Uso del sistema.
14. Realización de los diagramas de paquetes.
15. Realización de los diagramas de clases.
16. Descripción de las clases del diagrama.
17. Aplicación de métricas y métodos que serán usados para medir la calidad de los artefactos obtenidos.

Para la realización del trabajo fue necesario la utilización de los siguientes métodos y técnicas de investigación:

## **Métodos Teóricos:**

*Histórico-Lógico:* Utilizado para el estudio del procesos que se llevan a cabo en la FGR referentes a las quejas y el proceso de desarrollo del software.

*Analítico-Sintético:* Permite planificar toda la investigación logrando llegar a conclusiones prácticas a partir de la información que se procesa.

*Modelación:* Para la creación de modelos de negocio y otros que ayudarán a tener una vista más real de estos procesos en la FGR.

## **Métodos Empíricos:**

*Entrevista:* Permite realizar un profundo análisis de las necesidades del cliente para obtener información valiosa acerca del negocio y los requisitos en el desarrollo del software.

El presente trabajo de diploma está compuesto por 4 capítulos estructurados de la siguiente manera:

En el **Capítulo 1** se realiza la **Fundamentación Teórica** a través del estudio de los procesos referentes a Quejas y Reclamaciones que tienen lugar en la fiscalía llegando a conceptualizar cada uno de los términos utilizados. Por otro lado se tratan aspectos relacionados con el proceso de desarrollo de software haciendo un recorrido por las actividades a realizar para lograr el objetivo propuesto.





# *Introducción*

---

En el **Capítulo 2** se realiza el **Modelado del Negocio** describiendo y modelando los procesos de negocio mediante los Diagramas de Procesos de Negocio (DPN). Concluyendo el capítulo, se aplican métricas y métodos para validar los resultados obtenidos.

En el **Capítulo 3** se presenta la **Especificación de Requisitos** teniendo en cuenta las restricciones o políticas a cumplir por el sistema. Realizando la validación de los resultados obtenidos se aplican técnicas y métricas para medir la calidad de los requisitos identificados.

En el **Capítulo 4** se presenta el **Diseño** a través de los diagramas de paquetes y de clases. Se muestran además, una serie de vistas que formarán parte de la interfaz de usuario correspondiente a cada proceso logrando una detallada descripción de la arquitectura. Se aplican métricas para validar los resultados obtenidos.



## CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA

### 1.1 Introducción

En el presente capítulo se abordan aspectos teóricos y técnicos necesarios para realizar satisfactoriamente el análisis y diseño de los procesos referentes a quejas y reclamaciones del área PDC en la FGR. Se especifican los conceptos necesarios para el entendimiento de cada uno de los procesos pertinentes. Se abordan además las tendencias internacionales relacionadas, software que informatizan estos procesos y la necesidad de realizar éste para brindar un mejor rendimiento a los procesos que tienen lugar en la FGR referentes a quejas y reclamaciones de la población. Se realiza también un estudio del Programa de Mejora (PM) que se lleva a cabo actualmente en la universidad con el objetivo de alcanzar el nivel 2 de CMMI. Para el desarrollo del trabajo se tiene como guía el ciclo de vida incluido en el PM y un modelo de desarrollo iterativo e incremental, así como las herramientas a utilizar, notación, lenguaje de modelado, patrones y formas de validación que llevará implícito la aplicación de métricas y técnicas que serán usadas para medir la calidad de los artefactos obtenidos.

### 1.2 Conceptos fundamentales

**FGR:** Órgano del Estado al que corresponde, como objetivos fundamentales, el control y la preservación de la legalidad, sobre la base de la vigilancia del estricto cumplimiento de la Constitución, las leyes y demás disposiciones legales, por los organismos del Estado, entidades económicas y sociales y por los ciudadanos; y la promoción y el ejercicio de la acción penal pública en representación del Estado. (1)

**Queja:** Inconformidad ciudadana sobre violación de derechos constitucionales, garantías legalmente establecidas e infracciones cometidas en actos o disposiciones de organismos y funcionarios del estado y de entidades económicas y sociales. (2)

**Reclamación:** Solicitud, peticiones de un derecho, que no entraña presunción de violación de la legalidad, y requiere orientación sobre la base de la norma jurídica establecida. (2)

**Promovente:** Persona natural que presenta una queja o reclamación.

**Impugnación:** Derecho que tiene el promovente y es expresado a través de la acción que realiza el mismo al no estar de acuerdo con la respuesta obtenida tras la tramitación de su queja.



## Capítulo 1. Fundamentación Teórica

---

**Despacho:** Estudio de la queja o reclamación que hace el fiscal una vez que ésta llega a sus manos. En este momento se toman decisiones sobre cómo proceder con su tramitación.

**Materia Penal:** “El Derecho Procesal Penal es, el conjunto de normas jurídicas emanadas del poder del Estado que ordenan el proceso, sea en su conjunto, sea en los actos particulares que lo integran y que tiene por finalidad aplicar el derecho material para restablecer la legalidad quebrantada”. (3)

**Materia Civil:** Conjunto de actos, dirigidos a la investigación y resolución de los asuntos civiles y de familia, a través de un método pre-establecido por la ley, a fin de proteger el ordenamiento jurídico y los derechos de los particulares. (4)

**Materia Laboral:** Materia que encierra las quejas o reclamaciones que están relacionadas con todo lo que se refiere al trabajo.

**Materia Contravencional:** Constituirá contravención la infracción de las normas o disposiciones legales que carece de peligrosidad social por la escasa entidad de sus resultados. Las contravenciones serán definidas en decretos que dictará el Consejo de Ministros. (5)

**Legislación:** Son todas las leyes o documentos vigentes que pueden ser consultados para resolver un caso determinado.

**Legalidad:** Todo lo que es legal, es decir, que está permitido ante la ley.

**Orientación Jurídica:** Consiste en orientar, mediante escrito con el formato de una carta, al quejoso o reclamante sobre como encauzar en el orden legal su pedimento o solicitud, la procedencia o no de su reclamación o asunto planteado, con fundamento en la norma jurídica que así lo establece. (2)

**Traslado Definitivo (TD):** Tipo de traslado que decide darle un fiscal a una queja o reclamación para que sea tramitada en el órgano que le corresponda. No espera copia de repuesta ni recibe más información del caso.

**Traslado Bajo Control (TBC):** Tipo de traslado en que la queja o reclamación se traslada a la Fiscalía Provincial o Municipal según corresponda para que sea tramitada. Se espera una copia de la respuesta que se le da al promovente.



## Capítulo 1. Fundamentación Teórica

---

**Respuesta verbal:** Intercambio directo con la persona que posibilita una comprensión mayor y más completa de todas las interrogantes de la persona. (2)

**Solución operativa:** Son aquellas quejas, que mediante la acción del fiscal se resuelven conforme a la pretensión interesada por el promovente, y que no necesariamente implica la elaboración de un pronunciamiento por escrito. (2)

**Pronunciamiento no violatorio:** Cuando el fiscal, luego de practicadas las diligencias del caso, comprueba que no se ha quebrantado la legalidad, ofrecerá respuesta al promovente, con el formato de una carta, debidamente fundamentada, que abarque con un lenguaje sencillo en el orden técnico, los planteamientos formulados. (2)

### 1.3 Tendencias internacionales de las quejas o reclamaciones

Las quejas y reclamaciones de la población sobre distintas situaciones que respectan a violaciones de sus derechos, es un asunto de carácter internacional. En casi todos los lugares del mundo existe un órgano o persona natural que las atiende. Existe un “defensor” en la mayoría de los países capitalistas, que recibe las quejas y las dirige al fiscal correspondiente para que éste investigue y adopte las medidas oportunas, en su caso, para subsanar las deficiencias expuestas. En estos países, es la Fiscalía General de Estado quien cuenta con servicios de Atención al Ciudadano que tiene entre sus funciones tramitar las reclamaciones formuladas por los ciudadanos por correo electrónico, haciéndolas llegar en su caso al órgano competente para resolverlas. Por otro lado en la Orden ECO/734/2004 del 11 de marzo, se plantea también que la presentación de las quejas y reclamaciones podrá efectuarse, personalmente o mediante representación en soporte papel o por medios informáticos, siempre que éstos permitan la lectura, impresión y conservación de los documentos.

En Cuba no sucede así en cuanto a muchos aspectos, las quejas no se reciben por correo electrónico ni por vía telefónica, sino que se envían por correo postal o se presentan personalmente en la FGR. No existe un defensor del pueblo encargado de estos asuntos, todo el proceso de tramitación de la queja es realizado por el fiscal u órgano según corresponda para esto se cuenta con un término de sesenta días desde el momento en punto en que fue recibida. La queja es estudiada y tramitada por el fiscal que corresponda según la materia que trate la misma obteniéndose como resultado el pronunciamiento



redactado por el fiscal que constituirá la respuesta del promovente, el cual tiene derecho a impugnar en caso de no estar de acuerdo con la misma.

## **1.4 Software existentes que informatizan el proceso Quejas o Reclamaciones**

A medida que se desarrolla la sociedad, las empresas, organismos e instituciones, crece el volumen de información que se maneja lo cual obliga a invertir mucho tiempo en ella surgiendo así la necesidad de gestionarla permitiendo su aprovechamiento de una manera más eficiente, rápida y organizada. Debido a esto es necesario realizar un estudio de las diferentes soluciones de gestión fiscal que tratan las quejas o reclamaciones de la población. A continuación se muestran algunos de ellos:

### La Rama Judicial de Puerto Rico

Sitio web a través del cual la Rama Judicial de Puerto Rico presta un servicio a todas las personas que acudan a cualquier sala del Tribunal de Primera Instancia o al Tribunal de Apelaciones de Puerto Rico, o a cualquier otra unidad del Poder Judicial, y no sea atendida por los jueces y las juezas, o por el personal del sistema judicial con el fin de informar y orientarlas en cuanto al modo de presentar una queja contra esos funcionarios. De la misma forma, un ciudadano(a) también puede presentar una queja en contra de cualquier abogado, abogada o fiscal. Se muestra solamente el procedimiento a seguir para presentar la queja. (6)

### ComplaintsBoard

Es un sitio web que presta servicios para la presentación de quejas que presente un ciudadano contra una empresa determinada de cualquier parte del mundo. Da la posibilidad de elegir la categoría de la queja a modo de clasificación logrando obtener la información más organizada. El sitio permite ver las últimas quejas presentadas por lo que se evidencia que los temas más frecuentes son los relacionados con préstamos y robos monetarios. (7)

### Sistema de Control de las Quejas de la Población

Es el sistema que se utiliza actualmente en la FGR desde el año 2005. El mismo automatiza el proceso de control de las quejas de la población y se encuentra diseñado para el trabajo en la esfera de PDC solamente. Brinda facilidades de búsqueda a los fiscales del estado de tramitación de un asunto determinado. Permite además, buscar información por varios criterios ofreciendo la posibilidad de dar



# Capítulo 1. Fundamentación Teórica

---

seguimiento a los planteamientos. Este sistema no satisface todas las necesidades de los fiscales ya que sólo permite el registro y la búsqueda de quejas en materia penal, laboral, civil, familia y otras dirigidas a la fiscalía.

Esta aplicación está desarrollada en el lenguaje de programación *Visual Basic 6*, usando como gestor de base de datos *Microsoft Office Access*, como generador de reportes *Crystal Reports 8.5* y se generó su instalación con *Install Shield Professional 7.0* de lo que se puede deducir que fue creado con herramientas privativas que no cumplen con la meta que se propone alcanzar la FGR y el país en general: alcanzar la independencia tecnológica.

Después de realizado el estudio de diferentes soluciones informáticas en la gestión del trabajo de las fiscalías en el mundo, se puede plantear que, ni La Rama Judicial de Puerto Rico ni Complaints Board tienen compatibilidad con el sistema fiscal cubano, ya que son puramente informativos logrando orientar al ciudadano sobre cómo presentar una queja. Dichos “sistemas” carecen de funcionalidades para realizar cualquier tipo de solicitud referente a quejas que pueda ser presentadas por los ciudadanos y por ende no satisfacen, ni las necesidades del ciudadano, ni las del fiscal en asuntos de tramitación y respuesta para el promovente. Se evidencia entonces que el alcance de la reclamación que se presente sobre presuntas violaciones de un derecho determinado, es pobre, además, las leyes por las que se rigen no son iguales a las aplicadas en Cuba, no se tramitan todos los datos necesarios ni abarcan procesos esenciales en este sentido.

## **1.5 Proceso de Desarrollo de Software**

Dado a que se tiene como objeto de estudio el proceso de desarrollo de software, es importante definir en qué consiste el mismo: Un proceso de desarrollo de software encierra una serie de actividades y resultados encaminados a producir un software. (8). Existen metodologías que guían el proceso de desarrollo, por ejemplo, en la primera fase del proyecto SIGEF, fue utilizado RUP (*Rational Unified Process*) como metodología de desarrollo de software que logra asistir durante todo el ciclo de vida del proyecto. Permite llevar una organización y control del proceso de desarrollo previendo la mitigación temprana de riesgos que puedan atentar sobre el resultado final del producto. La misma está definida como un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. (9)



# Capítulo 1. Fundamentación Teórica

---

Como parte del proceso de desarrollo de software para realizar este trabajo, será utilizado el ciclo de vida definido como parte del Programa de Mejora (PM) que se está llevando a cabo actualmente en la universidad, el cual posee fases e iteraciones que sugieren qué hacer, no cómo hacerlo. Una vez realizado el estudio preliminar correspondiente a la fase de inicio incluida en este ciclo de vida de proyectos, se decide adoptar el mismo para guiar el proceso de desarrollo de software de la segunda fase del proyecto por ser éste una “adaptación” y por ende, muy similar a la metodología de desarrollo utilizada en la primera fase -RUP-. El uso de esta guía de trabajo, permitirá manejar correctamente proyectos a largo plazo haciendo especial énfasis en la generación de artefactos bien documentados que facilitan la capacitación y transferencia del producto.

El PM está basado en el modelo CMMI con la contratación de los servicios de consultoría del SIE Center (*Software Industry Excellence Center*) del Tecnológico de Monterrey. El objetivo fundamental que persigue la UCI es llegar a una certificación internacional del nivel 2 de este modelo. Además, permitirá lograr un crecimiento en cuanto a capacidades y madurez que se enfoca en todos los procesos logrando así una mejora considerable en el ciclo de vida dentro del desarrollo de software, mayor productividad, presupuestos predecibles en los proyectos productivos y mayor calidad en los productos. (10)

## 1.5.1 Aspectos importantes de CMMI

### ¿Qué es CMMI?

CMMI constituye una vista integradora de mejora de procesos a través de múltiples disciplinas y provee una orientación para la calidad de procesos. Cuenta con dos enfoques para la mejora del proceso: la capacidad del proceso y la madurez de la organización. (11)

Los modelos de CMMI soportan cada enfoque con una representación. La capacidad del proceso tiene una *representación continua* para una sola área o conjunto de áreas de proceso mientras que la madurez de la organización según el PM está dada por una *representación escalonada* compuesta por 5 niveles de madurez para un conjunto definido de áreas de proceso de la organización. (11)

### Niveles y áreas de CMMI



## Capítulo 1. Fundamentación Teórica

Dado a que CMMI ubica sus áreas de procesos en 5 niveles es necesario entender qué es un *área de proceso*: ésta no es más que un conjunto de prácticas que cuando se llevan a cabo colectivamente, satisfacen un número de metas importantes para hacer la mejora en un área determinada. (11).

Para hacer este trabajo se tomará como referencia los niveles 1 y 2:

- ✓ Nivel 1 – Inicial: Proceso impredecible.

En este nivel se encuentran todas las empresas que no tienen procesos. No hay control sobre el estado actual del proyecto, el desarrollo del proyecto es completamente opaco. Si no se sabe el tamaño del proyecto ni cuánto llevas hecho, nunca se sabrá cuando terminar. (12)

- ✓ Nivel 2 – Administrado: Administración básica del proyecto.

En este nivel los proyectos se ejecutan y gestionan de acuerdo con los planes de proyecto. Se sabe cuánto trabajo está hecho y cuánto queda por hacer. Los elementos de trabajo se revisan con las personas involucradas y son controlados. (12)

Actualmente la UCI se encuentra insertada en el nivel 1 de CMMI. Para lograr el objetivo trazado - alcanzar el nivel 2- se están realizando una serie de mejoras en las diferentes áreas de procesos incluidas en el mismo como se muestra a continuación:

Área de Proceso	Nivel
Gestión de la Configuración	2
Gestión de Requisitos	2
Monitorización y control de proyecto	2
Planificación de Proyecto	2
Gestión calidad procesos y producto	2
Medición y Análisis	2
Gestión y acuerdo con proveedores	2

**Tabla 1 Áreas de proceso de CMMI. Nivel 2**

Específicamente se hará énfasis en el área que enmarca el análisis y diseño a realizar, la misma corresponde al nivel 2 de CMMI y es precisamente área de Gestión de Requisitos:





# Capítulo 1. Fundamentación Teórica

---

- ✓ Gestión de Requisitos: Tiene como propósito administrar los requisitos del proyecto e identificar inconsistencias entre dichos requisitos, los subproductos y planes del proyecto. (11)

## 1.5.2 Ciclo de vida del PM

La realización de este trabajo está basada en el ciclo de vida de proyectos incluido en el PM que define el Centro de Calidad para Soluciones Informáticas (CALISOFT) y en un modelo de desarrollo iterativo e incremental constituido por cuatro fases: Inicio, Elaboración, Construcción y Transición. Según este ciclo de vida, el proceso de desarrollo incluye las siguientes disciplinas: Estudio Preliminar, Modelado del Negocio, Requisitos, Análisis y Diseño, Implementación, Pruebas Internas, Pruebas de Liberación, Despliegue y Soporte. (13)

Para este trabajo servirán de guía las siguientes fases e iteraciones:

### Fase de Inicio

- ✓ Estudio Preliminar.

### Fase de Elaboración

- ✓ Modelado de negocio: En esta fase se generan artefactos como: Descripciones de procesos de negocio, Mapas de procesos de negocio y el Modelo conceptual correspondiente.
- ✓ Requisitos: Los artefactos obtenidos en esta fase son: Especificación de Requisitos de Software y Descripción de Requisitos de Software.
- ✓ Análisis y Diseño: Se obtiene el Modelo de Diseño como artefacto correspondiente a esta disciplina el cual incluye .los Diagramas de clases del diseño, Diagrama de Paquetes y Descripción de las clases.

A continuación se definen algunos conceptos importantes de recalcar en cada una de las disciplinas en que se estará trabajando:

### **Modelado de negocio**

Es la disciplina destinada a comprender cómo funciona el negocio que se desea automatizar para tener garantías de que el software desarrollado va a cumplir su propósito. Será realizado centrado en los procesos relacionados con quejas o reclamaciones de la población. (13)

### **Requisitos**



## Capítulo 1. Fundamentación Teórica

---

El esfuerzo principal en la disciplina de Requisitos es desarrollar un modelo del sistema que se va a construir. Incluye un conjunto de requisitos funcionales y la especificación de requisitos no funcionales. (13)

*¿Qué es un requisito?*

Según el PM un requisito es una condición o capacidad requerida por el usuario para resolver un problema o alcanzar un objetivo, debe ser cumplida o poseída por un producto o por un componente de producto para satisfacer un contrato, estándar, especificación, u otro documento impuesto formalmente. (14)  
Necesidad del usuario o característica, función, estado o atributo de un sistema que puede ser percibida externamente. (15)

Los requisitos clasifican en Requisitos Funcionales y No Funcionales:

*Requisitos funcionales (RF):* definen las funciones que el sistema será capaz de realizar, describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Es importante que se describa el *¿Qué?* y no el *¿Cómo?* se deben hacer esas transformaciones. Además, son independientes de las tecnologías usadas por el producto. (16)

*Requisitos no Funcionales (RNF):* son aquellos requisitos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento. (17)

En vistas a mejorar la calidad del software se tienen en cuenta algunos aspectos de la ingeniería de requisitos de modo que los resultados obtenidos reflejen en el software las verdaderas necesidades del cliente. A continuación se explica en qué consiste:

*Ingeniería de Requisitos*

Según Pressman la Ingeniería de Requisitos (IR) ayuda a los ingenieros de software a entender mejor el problema en cuya solución trabajarán. Incluye el conjunto de tareas que conducen a comprender cuál será el impacto del software sobre el negocio, qué es lo que el cliente quiere y cómo interactuarán los usuarios finales con el software.

*Fases de la Ingeniería de Requisitos*



## Capítulo 1. Fundamentación Teórica

---

La IR se divide en etapas lógicamente bien definidas, las cuales especifican los pasos, las tareas y las técnicas que se deben emplear para interactuar con los clientes y especificar correctamente el sistema. Es importante mantener una fluida comunicación entre los involucrados en el desarrollo del proyecto y una guía para especificar en detalles el sistema solicitado. A continuación se explica en qué consisten cada una de las fases:

- ✓ Obtención de requisitos.

Es la actividad mediante la cual, el equipo de desarrollo de un sistema de software extrae, de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema. (18).

Mediante la captura de requisitos los analistas deben ser capaces de determinar cuáles son las necesidades del cliente que debe satisfacer el sistema para lo cual le es imprescindible una buena comunicación con el mismo.

- ✓ Análisis de requisitos.

Una vez los requisitos están recopilados, se estudia cada requisito en relación con el resto, se examinan los requisitos según su consistencia, completitud y ambigüedad, y se clasifican en base a las necesidades de los clientes/usuarios. (18)

- ✓ Especificación de requisitos.

Una especificación puede ser un documento escrito, un modelo gráfico, un modelo matemático formal, una colección de escenarios de uso, un prototipo o una combinación de lo anterior. A través de la especificación de requisitos se puede negociar concretamente lo que el sistema debe hacer y cumplir para evitar retrasos y costes agregados por la mitigación de errores. Es el punto de partida de la estimación de costo, tiempo y esfuerzo del proyecto y tiene una incidencia marcada en los procesos de gestión de la calidad del producto a entregar. (19)

- ✓ Validación de requisitos.

La validación de requisitos tiene como objetivo demostrar que la especificación de los requisitos define realmente el sistema que el usuario necesita o el cliente desea. Examina las especificaciones para asegurar que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin inconsistencias, sin omisiones, que los errores detectados hayan sido corregidos, y que el resultado del trabajo se ajusta a los estándares establecidos para el proceso, el proyecto y el producto (18)

- ✓ Administración de requisitos.



# Capítulo 1. Fundamentación Teórica

---

Tiene como objetivo: “Administrar los requisitos del proyecto e identificar inconsistencias entre dichos requisitos, los subproductos y planes del proyecto”. (11)

## **Diseño**

Durante esta disciplina es modelado el sistema de forma tal soporte todos los requisitos, incluyendo los no funcionales. Son desarrollados los diagramas de clases y de paquete. (13)

Es importante centrar el mayor esfuerzo en esta etapa, tener como guía soluciones que han sido probadas en casos anteriores con resultados satisfactorios y que sean similares al sistema que se esté construyendo con el objetivo de lograr mejores resultados ya que sin un buen diseño es imposible garantizar la calidad del producto final.

En cada una de estas disciplinas se generan artefactos, los más significativos para realizar este trabajo están incluidos en la fase de elaboración. Para obtener estos artefactos con la calidad requerida, dependiendo del caso es necesario aplicar un conjunto de técnicas que permitan obtener resultados realmente útiles. En el siguiente epígrafe se hace énfasis en las mismas.

## **1.6 Conceptualización de las técnicas utilizadas**

Es necesario aclarar que las técnicas que a continuación se muestran fueron definidas una vez realizado el estudio preliminar correspondiente a la fase de inicio del ciclo de vida antes mencionado.

### **Técnicas para el Modelado del negocio**

*Observación:* Es utilizada para capturar visualmente y registrar por escrito el desenvolvimiento de las actividades habituales. (20)

*Entrevista:* Es una técnica complicada pues depende de la habilidad del entrevistador, ya que es determinante seleccionar bien a los entrevistados para obtener la mayor cantidad de información importante en un período de tiempo limitado. Es muy aceptada y permite acercarse al problema de una manera natural. Abarca cuatro pasos: identificación de los entrevistados, preparación de la entrevista, realización de la entrevista y documentación de los resultados.

*Tormentas de ideas:* Es una técnica cuyo objetivo es la generación de ideas en un ambiente libre de críticas o juicios. Puede ayudar a generar una gran variedad de vistas del problema y a formularlo de



## Capítulo 1. Fundamentación Teórica

---

diferentes formas. Suele estar formada por un número de cuatro a diez participantes, uno de los cuales es el jefe de la sesión, encargado más de comenzar la sesión que de controlarla.

*Análisis de documentos:* Constituye una forma de investigación cuyo objetivo es la captación, evaluación, selección y síntesis de los mensajes subyacentes en el contenido de los documentos, a partir del análisis de sus significados, a la luz de un problema determinado. Así, contribuye a la toma de decisiones, al cambio en el curso de las acciones y de las estrategias.

En la disciplina de Requisitos se aplicó también la combinación de las técnicas anteriores – exceptuando el análisis de documentos- con el objetivo de hacer una correcta captura de requisitos que permita incluir en el sistema lo que realmente será de útil para el cliente. Siendo la tormenta de ideas muy útil al comienzo del proceso cuando aún los requisitos resultaban difusos.

### **1.7 Definiciones teóricas de los patrones utilizados**

Los siguientes patrones son utilizados para la realización del presente trabajo en las fases correspondientes, los mismos fueron definidos en el estudio preliminar correspondiente a la fase de inicio del ciclo de vida antes mencionado.

#### ¿Qué es un patrón?

Los patrones son soluciones simples compuestas por una pareja problema/solución, fundamentadas en la experiencia para problemas específicos y comunes. Se ha demostrado que funcionan y pueden emplearse en diferentes contextos. (21)

#### ¿Para qué sirven los patrones?

Los patrones permiten establecer un vocabulario de diseño común entre los desarrolladores ayudando así a identificar los errores y obstáculos comunes que ocurren a la hora de crear un sistema. Además, son arquitecturas comprobadas para construir software orientado a objetos que sea flexible y se pueda mantener. A continuación se muestran una serie de patrones que fueron utilizados a lo largo del desarrollo de las disciplinas en cuestión:

#### ***Patrones aplicados en el modelado de los procesos de negocio (22):***

Existen patrones definidos bajo el estándar de BPMN que se tienen en cuenta para el modelado.



## Capítulo 1. Fundamentación Teórica

---

### Patrones Básicos de Control de Flujo

- ✓ Secuencia

Un diagrama de procesos de negocio ilustra este patrón como una serie de actividades conectadas por flujos de secuencia. Se requiere cuando hay una dependencia entre dos actividades, de tal forma que una actividad no pueda iniciarse antes de que otra haya terminado.

- ✓ Distribución en Paralelo

La distribución en paralelo es necesaria cuando dos o más actividades deben ejecutarse de forma concurrente o en paralelo. Este patrón se encuentra en un punto en el flujo de trabajo en donde un hilo de control del proceso debe dividirse en múltiples hilos que deben ser ejecutados en paralelo, y por ende, permitiendo la ejecución de actividades simultáneamente y en cualquier orden.

- ✓ Selección Exclusiva

Ocurre cuando en un punto del flujo de trabajo se escoge sólo una de varias ramas del proceso, generalmente esta decisión se toma basándose en datos de control del flujo de proceso.

### Patrones de Sincronización y Enrutamiento Avanzada

- ✓ Selección Múltiple

Este patrón asume que una de las alternativas es seleccionada y ejecutada. Sin embargo, algunas veces es menester usar una opción en la cual se permita escoger un subconjunto de alternativas de un conjunto de opciones posibles. Por ello, se encuentra el patrón de selección múltiple (inclusiva). La selección múltiple se encuentra en un punto del flujo de trabajo en donde, basándose en una decisión o en los datos de control del mismo, se escoge una o más ramas activando uno o más caminos del proceso.

- ✓ Mezcla Múltiple o Serialización

La Mezcla Múltiple es un patrón del flujo de trabajo en donde dos o más ramas se unen sin sincronización. Si se activa más de una rama, posiblemente de forma concurrente la actividad que sigue a la serialización se ejecuta una vez por cada rama que ha sido activada.

### ***Patrones o técnicas aplicadas en la disciplina de requisitos:***



# Capítulo 1. Fundamentación Teórica

---

## Técnicas

### ✓ Plantillas o patrones

Esta técnica, tiene por objetivo el describir los requisitos mediante el lenguaje natural pero de una forma estructurada. Una plantilla es una tabla con una serie de campos y una estructura predefinida que el equipo de desarrollo va cumplimentando usando para ello el lenguaje del usuario. Las plantillas eliminan parte de la ambigüedad del lenguaje natural al estructurar la información; cuanto más estructurada sea ésta, menos ambigüedad ofrece. (23)

## Patrones de casos de uso

### *Múltiples actores*

Para capturar la concordancia entre actores manteniendo roles separados.

### ✓ 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.

Este patrón brinda la posibilidad de realizar el modelado del sistema de una manera más eficiente, lo cual es de gran importancia, pues este modelo es una entrada fundamental para realizar el diseño del sistema. Los patrones de casos de uso juegan un papel importante ya que permiten ahorrar tiempo, coste y se hace más fácil el trabajo.

### ***Patrones aplicados en la disciplina de diseño:***

Los patrones de diseño proponen soluciones a un problema dentro de un contexto dado. Son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos. Una vez entendido el funcionamiento de los patrones, los diseños serán mucho más flexibles, y reutilizables. (24)

Los **patrones GRASP** resultaron ser aplicados en esta disciplina, su acrónimo significa: patrones generales de software para asignar responsabilidades, del inglés: *General Responsibility Assignment*



## Capítulo 1. Fundamentación Teórica

---

*Software Patterns*. Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.

Larman enuncia nueve patrones GRASP, se pueden destacar 5 patrones principales y 4 patrones adicionales, entre estos últimos están los conocidos como: Polimorfismo, Indirección, Fabricación Pura y No hables con Extraños. (25)

### Patrones GRASP principales:

#### ✓ Patrón Experto:

Solución: Asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad.

Problema: ¿Cuál es el principio fundamental en virtud del cual se asignan las responsabilidades en el diseño orientado a objetos?

Este patrón se usa más que cualquier otro al asignar responsabilidades. El patrón Experto ofrece una analogía con el mundo real. (26)

#### ✓ Patrón Creador:

Solución: Asignarle a la clase B la responsabilidad de crear una instancia de la clase A en uno de los siguientes casos:

B agrega los objetos de A.

B contiene los objetos de A.

B registra las instancias de los objetos A.

B utiliza específicamente los objetos A.

B tiene los datos de inicialización que serán transmitidos a A cuando este objeto sea creado (así B es un Experto respecto a la creación de A).

Problema: ¿Quién debería ser el responsable de crear una nueva instancia de alguna clase? El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy





## Capítulo 1. Fundamentación Teórica

---

frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que debemos conectar con el objeto producido en cualquier evento. (26)

✓ *Patrón Bajo Acoplamiento:*

Solución: Asignar una responsabilidad para mantener bajo acoplamiento.

Problema: ¿Cómo dar soporte a una dependencia escasa y a un aumento de la reutilización?

El Bajo acoplamiento es un principio que debemos recordar durante las decisiones del diseño: es la meta principal que es preciso tener siempre presente. (26)

✓ *Patrón Alta cohesión:*

Solución: Asignar una responsabilidad de modo que la cohesión siga siendo alta.

Problema: ¿Cómo mantener la complejidad dentro de límites manejables?

Alta cohesión es un principio de debemos tener presente en todas las decisiones de diseño: es la meta principal que ha de buscarse en todo momento. Es un patrón evaluativo que el desarrollador aplica al valorar sus decisiones de diseño. (26)

✓ *Patrón Controlador:*

Solución: Asignar la responsabilidad del manejo de mensajes de los eventos del sistema a una clase que represente alguna de las siguientes opciones:

- El sistema global.
- La empresa u organización global.
- Algo activo en el mundo real que pueda participar en la tarea.
- Un manejador artificial de todos los eventos del sistema de un caso de uso (controlador de casos de uso).

Problema: ¿Quién debería encargarse de atender un evento del sistema?

Otros de los patrones que resultaron ser aplicados en esta disciplina son los **Patrones GOF**, éste es el acrónimo de Gang of Four (en español, banda de los cuatro), se clasifican de acuerdo a su propósito en:



# Capítulo 1. Fundamentación Teórica

---

## ✓ Patrones de creación

Se encargan de las formas de crear instancias de objetos. El objetivo de estos patrones es abstraer el proceso de instanciación y ocultar los detalles de cómo los objetos son creados o inicializados. (27)

Abstract Factory: Permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí y haciendo transparente el tipo de familia concreta que se esté usando.

Singleton: Garantiza que una clase sólo tenga una instancia, y proporciona un punto de acceso global a ella.

## ✓ Patrones estructurales

Se ocupan de la relación entre clases, la combinación de objetos y la formación de estructuras de mayor complejidad. Describen como las clases y objetos pueden ser combinados para formar grandes estructuras y proporcionar nuevas funcionalidades. Estos objetos adicionados pueden ser objetos simples u objetos compuestos. (27)

Composite: Permite que los clientes traten de manera uniforme a los objetos individuales y a los compuestos.

Decorator: Añade funcionalidad a una clase dinámicamente.

## 1.8 Métricas

La IEEE (IEEE, 1990) define como métrica: “una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado”.

### Métricas para los requisitos

Las métricas juegan un papel importante, en este caso, brindarán información sobre la calidad y validez de los requisitos. Se estudiaron métricas como el *Grado de validación de los requerimientos*, útil para medir la corrección en la definición de los requerimientos, el *Promedio de requerimientos por necesidades* con la que se verifica que las necesidades del cliente se encuentren cubiertas por los requisitos especificados y la métrica *Estabilidad de los requerimientos*, resultó ser la aplicada con este fin, evitándose así adiciones o supresiones en los mismos que impliquen posteriores modificaciones en las funcionalidades principales del producto, por lo que se decidió utilizar esta última para validar los requisitos identificados.



## Capítulo 1. Fundamentación Teórica

---

### Métricas para la Calidad de los Casos de Uso (CU)

Consiste en la aplicación de preguntas agrupadas bajo categorías relativas a características de calidad, con el fin de comprobar defectos que puedan estar contenidos en los Casos de Uso. (28). Esta métrica está compuesta por 4 atributos:

- ✓ *Compleitud*: 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 de negocio.
- ✓ *Complejidad*: Grado de claridad en la presentación de los elementos que describen el contexto y la claridad del sistema.

### Métricas para evaluar el Diseño de Software

Para medir la calidad del diseño se utilizaron métricas básicas inspiradas en el estudio de la calidad del diseño orientado a objeto referenciadas por Pressman teniendo en cuenta que este estudio brinda un esquema sencillo de implementar y que a la vez cubre los principales atributos de calidad de software. Las métricas escogidas para la validación del diseño fueron Métricas Orientadas a Clases (Tamaño de clase) y la Métrica Árbol de Profundidad de Herencia (APH). A continuación se explica en qué consiste cada una de ellas:

- ✓ *Árbol de Profundidad de Herencia (APH)*

La métrica APH propuesta por Chidamber y Kemerer (CK), se define como la longitud máxima desde el nodo hasta la raíz del árbol, el nodo es una clase hija que hereda de una clase, y así respectivamente hasta llegar a la raíz. A medida que esa longitud va creciendo, entonces se van heredando más operaciones y atributos por las clases hijas y la complejidad del módulo aumenta haciéndose más difícil predecir el comportamiento de las clases que ocupan niveles inferiores. (29)

*Definición*: mide el máximo nivel en la jerarquía de herencia. Es la cuenta directa de los niveles en la jerarquía de herencia. En el nivel cero de la jerarquía se encuentra la clase raíz.



## Capítulo 1. Fundamentación Teórica

*Propósito:* medida de la complejidad de una clase, complejidad del diseño y el potencial reuso. Esto es debido a que cuanto más profunda se encuentra una clase en la jerarquía, mayor será la probabilidad de heredar un mayor número de métodos.

Si los valores de APH son grandes, entonces se garantiza que se reutilice gran cantidad de código; pero al mismo tiempo hace que el diseño sea más complejo, esto provoca un mayor acoplamiento entre las clases. (29)

- ✓ *Métricas orientadas a clases.*

### Tamaño de clase (TC):

Con el objetivo de comprobar el adecuado diseño de las clases y el nivel de reutilización de las mismas se aplicó la métrica del TC. Será aplicada sólo para las principales clases de los módulos definidos en el sistema.

El tamaño general de una clase se puede determinar siguiendo los planteamientos a continuación:

- ✓ El número total de operaciones (tanto operaciones heredadas como operaciones privadas de la instancia) que están encapsuladas dentro de la clase.
- ✓ El número de atributos (tanto atributos heredados como atributos privados de la instancia) que están encapsulados en la clase.

Con la existencia de grandes valores de TC se demostrará que una clase puede tener demasiada responsabilidad, lo cual reduciría la reutilización de la clase y hará complicada la implementación y las pruebas. De forma contraria sucede si los valores TC son de menor valor. Finalmente se calcula los promedios correspondientes a los diferentes valores para tener una estimación general del sistema.

Es necesaria una evaluación de las métricas mediante los umbrales:

**Tabla 2 Clasificación de las Clases**

Clasificación	Valores de Umbrales
Pequeño	$\leq 20$
Mediano	$20 < \leq 30$
Grande	$> 30$



## 1.9 Herramientas propuestas por el Programa de Mejora

El PM propone el uso de algunas herramientas para el desarrollo de los artefactos correspondientes a cada una de las disciplinas:

### *Herramientas CASE*

Las herramientas CASE (Ingeniería de Software Asistida por Computadoras, del inglés, *Computer Aided Software Engineering*), son aplicaciones que facilitan el desarrollo de software, reduciendo el esfuerzo, el costo y el tiempo, además de estructurar la documentación asociada a los artefactos generados. (30).

Las herramientas CASE de desarrollo de software más utilizadas en el mundo de la producción de software son: *Enterprise Architect*, *Rational Rose* y *Visual Paradigm*.

### Herramienta para el modelado de procesos de negocio

- ✓ *Visual Paradigm (Versión 3.4)*

Entre las herramientas que propone el PM para el modelado se encuentran: *Star UML*, *Enterprise Architect* y *Visual Paradigm*. Resultó ser ésta última la seleccionada para el modelado de los procesos en cuestión ya que es una herramienta profesional y logra llevar el ciclo de vida completo del desarrollo de software. Es más amigable que otras herramientas CASE y flexible en la medida de lo posible, mucho más fácil de usar, sus componentes se encuentran relacionados, por lo que se hace muy fácil la creación de cualquier tipo de diagrama. (31). Además facilita el modelado de los diagramas necesarios durante el análisis y el diseño. Es fácil de instalar y actualizar, permite generar DPN y Diagramas de Clases, lo cual será muy útil para llevar a cabo la primera etapa del proyecto. (32) También se sustenta la elección de la misma en el hecho de en la UCI se encuentra la licencia para el uso de dicha herramienta lo que facilita su uso a pesar de ser un software propietario.

### Herramienta para el prototipado

- ✓ *Axure RP (Versión 5.5)*

Axure RP es una aplicación para crear prototipos y especificaciones muy precisas para páginas web. Permite componer la página web visualmente, añadiendo, quitando y modificando los elementos con suma facilidad. Es fácil de entender y de aprender a usar. Está enfocado justamente en el punto más crítico de



los proyectos de software que es la satisfacción con el cliente, logrando un excelente puente entre el análisis y el diseño de todos los proyectos. (33)

## 1.10 Notación y lenguaje de modelado

*Notación para el modelado de procesos de negocio:*

Para realizar el modelado de procesos de negocio se utilizó BPMN (*Bussiness Proccess Modeling Notation*) como notación que logra ajustarse a las necesidades que presenta el proyecto para proveer un modelado de procesos que resulte fácilmente entendible por todos los usuarios del negocio, desde los analistas hasta los desarrolladores técnicos, y no por último serán menos importantes los clientes que estarán manejando el proceso de negocio. BPMN define un Diagrama de Procesos de Negocio (DPN) basado en una técnica de diagramas de flujo que será utilizado para modelar los procesos de negocio. Esta notación de modelado tiene actualmente una alta aceptación en proyectos de software, a pesar de ser relativamente nueva, cada día crece su utilización ya que permite describir los procesos de negocio tal y como ocurren en tiempo real.

*Lenguaje de Modelado*

El surgimiento de numerosos lenguajes de modelado está dado por la necesidad de la comunidad del software de comunicar sus modelos. El Lenguaje Unificado de Modelado (*Unified Model Language, UML*) es un lenguaje gráfico para visualizar, especificar, construir, documentar y comunicar los artefactos de un sistema de software. (34).

Entre sus funciones principales se encuentran las siguientes:

- ✓ *Visualizar:* Permite expresar de una forma gráfica un sistema de forma que otro lo pueda entender.
- ✓ *Especificar:* Permite especificar cuáles son las características de un sistema antes de su construcción.
- ✓ *Construir:* A partir de los modelos especificados se pueden construir los sistemas diseñados.
- ✓ *Documentar:* Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Por tanto, se decidió utilizar **UML** como lenguaje para la elaboración del modelo conceptual correspondiente ya que es un lenguaje de modelado ajustable para proyectos de software de gran envergadura, está compuesto por diversos elementos gráficos que se combinan para conformar



diagramas cuya finalidad es presentar diversas perspectivas del sistema, a las cuales se le conoce como modelo. Un modelo UML describe lo que supuestamente hará el sistema, pero no dice cómo implementarlo. (35)

Posee formas de modelar conceptos como son los procesos de negocio y funciones de sistema, además de aspectos concretos como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables. (36)

## 1.11 Arquitectura del sistema

La arquitectura de un sistema es tan importante como cualquier aspecto relacionado con la construcción de un software. A grandes rasgos incluye los componentes principales que conformarán el sistema, la conducta de los mismos, así como la forma en que interactúan para alcanzar la misión del sistema. La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución. (37)

### Plataforma de desarrollo

El sistema se desarrollará en plataforma *Linux Apache Postgre PHP* (LAPP). El principal motivo de esta decisión es la política de migración hacia software libre que tiene la FGR y por ser además la estrategia del país en alcanzar la soberanía tecnológica. Esta plataforma tiene gran prestigio internacional, en ella se desarrollan un gran número de las aplicaciones web hoy en día. Para el desarrollo de la solución de software se utilizará el *Netbeans* 6.8 como Entorno de Desarrollo Integrado (*IDE*, siglas en inglés), ya que ofrece un amplio completamiento de código para PHP en el uso del framework *symfony* que fue seleccionado para el desarrollo por su amplia cobertura de funciones para generar componentes reutilizables, su capacidad de interactuar fácilmente mediante los *helpers* con Ajax, lenguaje del lado del cliente, y que a su vez propone el uso de *Propel*, otro framework especializado en el mapeo de objetos siendo el resultado de esta integración la generación de componentes reutilizables desde la capa de acceso a datos hasta la vista de la aplicación. (13)



## *Capítulo 1. Fundamentación Teórica*

---

### **1.12 Conclusiones parciales**

Luego de un estudio realizado del Proceso Quejas o Reclamaciones es evidente la necesidad que existe por parte de las fiscalías del país de contar con un sistema que le brinde una mayor fuerza procesal que logre solucionar el problema que originó esta investigación.

Con este fin se realizó un estudio que fundamenta la elección del lenguaje UML, la notación BPM, herramientas para el prototipado y para el modelado (Axure y Visual Paradigm respectivamente) así como otros aspectos del proceso de desarrollo de software que se logran ajustar a las necesidades del proyecto. Como resultado de este trabajo se obtendrá una primera parte de los artefactos definidos en el PM. De este modo se dará un paso de avance en la construcción del software que según las expectativas existentes podrán satisfacer las necesidades del cliente.





## Capítulo 2. Modelado del Negocio

---

### CAPÍTULO II. MODELADO DEL NEGOCIO

#### 2.1 Introducción

En este capítulo se dará a conocer lo relacionado con el **Modelado de negocio** describiendo y modelando los procesos de negocio mediante los Diagramas de Procesos de Negocio (DPN) utilizando BPMN, realizando una previa identificación de los mismos. Se generan entregables como el modelo conceptual para lo que se utilizó el lenguaje UML en la confección de los diagramas, los mapas de procesos y las descripciones de procesos de negocio correspondientes. Se aplican técnicas, patrones y métricas para validar los artefactos generados, además de realizar las validaciones con el cliente.

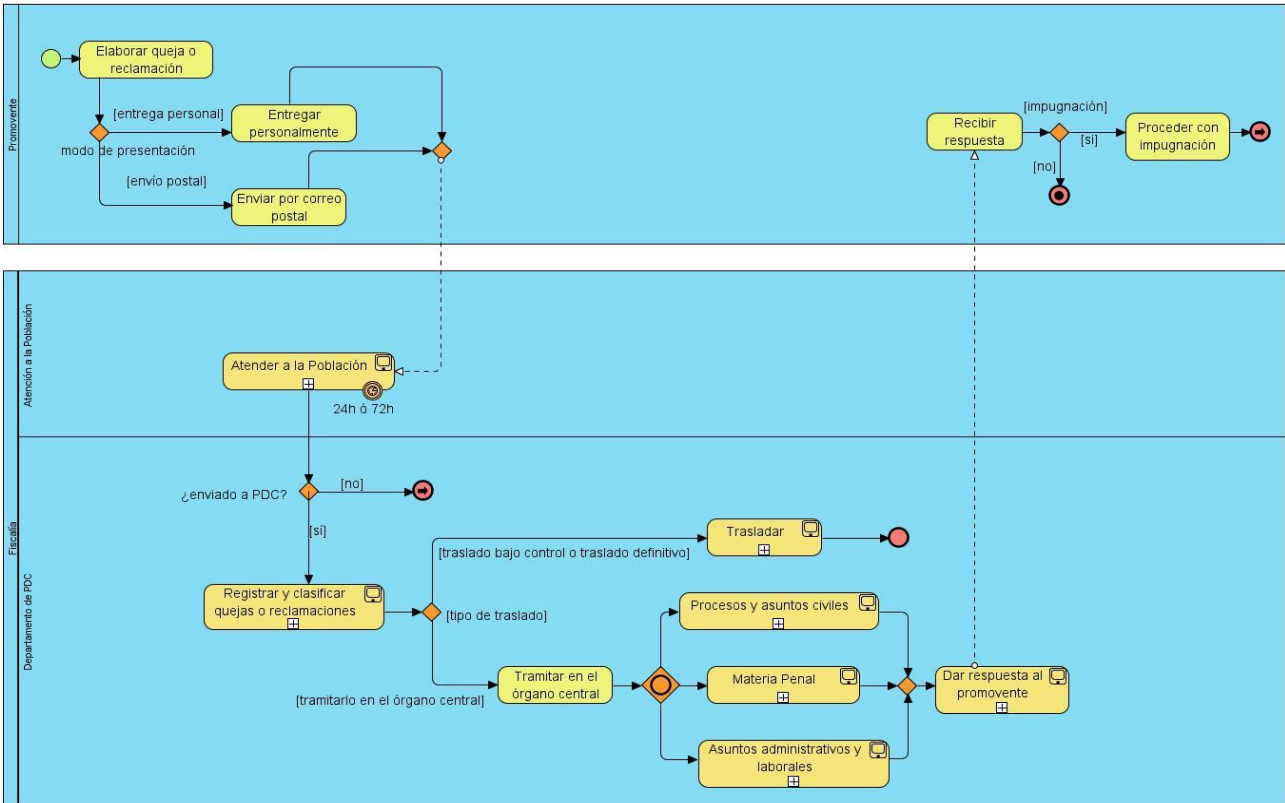
#### 2.2 Modelado del Negocio

Seguidamente se procede a describir los principales procesos de negocio referentes a Quejas y Reclamaciones que se llevan a cabo en el departamento PDC de la FGR. De esta manera se pretende lograr un entendimiento común del negocio por parte de los clientes y los analistas una vez obtenido el modelado con BPMN. Se mostrará un caso muy puntual para ejemplificar el modelado, por lo que el resto de los diagramas y sus descripciones se encuentran en el expediente del proyecto. El diagrama que se muestra a continuación representa la tramitación de las quejas de forma general. De aquí se derivan una serie de subprocesos en los que se modelan las especificidades, estos son:

- ✓ Atender a la Población. (*Ver Anexo 1*).
- ✓ Registrar y clasificar Quejas o Reclamaciones. (*Ver Anexo 2*).
- ✓ Trasladar Queja o Reclamación. (*Ver Anexo 3*).
- ✓ Procesos y Asuntos Civiles. (*Ver Anexo 4*).
- ✓ Materia Penal. (*Ver Anexo 12*).
- ✓ Asuntos Administrativos y Laborales. (*Ver Anexo 6*).
- ✓ Dar respuesta al promovente. (*Ver Anexo 5*).



## Capítulo 2. Modelado del Negocio



**Figura Nº 1 Diagrama de Procesos de Negocio de las Quejas o Reclamaciones General**

En el proceso mostrado anteriormente se modela cómo ocurre de manera general la tramitación de las quejas o reclamaciones en la FGR. Como se puede apreciar la queja es elaborada por el promovente quien puede hacer entrega de la misma personalmente o enviarla por correo postal, en ambos casos es recibida en el Dpto. Atención a la población de la fiscalía. Luego, es enviada al Dpto. de PDC donde se registra y se clasifica la queja con el fin de decidir si se va a trasladar a otro lugar para que sea tramitada (puede ser a otro Dpto. dentro de la propia fiscalía o al órgano que corresponda tramitar el asunto) o si se va a tramitar en el propio Órgano Central. En caso de que la queja sea trasladada –el fiscal decide que es un TBC o un TD- se realiza el traslado correspondiente y en caso de que se tramite en el Órgano Central se trata la queja dependiendo de la materia con que se relacione la misma: Materia Penal, Procesos y Asuntos Civiles o Asuntos Administrativos y Laborales. Finalmente el fiscal le da respuesta al promovente quien tiene derecho a impugnarla o no dependiendo de su inconformidad con la respuesta recibida.



## Capítulo 2. Modelado del Negocio

---

### 2.2.1 Resultados de la aplicación de las técnicas

Para entender y luego modelar el negocio referente a los procesos de quejas o reclamaciones en la FGR fue necesario aplicar la combinación de varias técnicas para comprender el negocio y obtener de esta manera información consistente sobre los procesos de negocio en cuestión y así delimitar el alcance del producto a obtener. En el capítulo 1 se explicó en qué consisten cada una de las técnicas que a continuación se muestran por lo que en este momento se describen solamente los resultados palpables luego de la aplicación de cada una de ellas:

*Observación:* Se practicó esta técnica en cada uno de los encuentros con los fiscales para tener constancia por escrito de las acciones realizadas. Fue muy útil para entender los procesos de negocio ya que el analista pudo sentir de cerca cómo ocurrían los procesos en tiempo real, en este caso, viendo la forma de tramitación de las quejas en la propia FGR.

*Tormentas de ideas:* Se realizaron talleres con especialistas de la FGR y analistas del proyecto para obtener más información sobre los procesos en cuestión.

*Análisis de documentos:* Se realizó un estudio de la legislación vigente relacionada con estos procesos para lograr un correcto modelado del negocio, para determinar las políticas y condiciones que se deben cumplir para la gestión y control de documentos para el proceso quejas y reclamaciones.

*Entrevistas:* Se realizaron varias entrevistas -más de diez- con los fiscales implicados con el objetivo de obtener la mayor cantidad posible que información sobre el funcionamiento del negocio. Los resultados de cada una de las entrevistas realizadas quedaron documentados mediante las minutas de reunión.

### 2.2.2 Patrones de negocio

En el primer capítulo se explicó en qué consistía teóricamente cada uno de los patrones de negocio que a continuación se ven aplicados a lo largo del modelado de procesos:

#### *Patrones Básicos de Control de Flujo*

##### ✓ *Secuencia*

Se ve aplicado en el subproceso Estudiar Queja o Reclamación (*Ver Anexo 10*) donde se evidencia la dependencia entre las actividades ya que para enviar el cumplimiento de la disposición a la fiscalía, es necesario recibir el pronunciamiento y responderlo en el término de tiempo establecido:



## Capítulo 2. Modelado del Negocio



Figura Nº 2 Patrón Secuencia

### ✓ Distribución en Paralelo

Se ve reflejado en el subproceso Estudiar Queja o Reclamación (Ver Anexo 10) donde las actividades Notificar al Infractor y Dar cuenta al superior se ejecutan concurrentemente:



10:49

Figura Nº 3 Patrón Distribución en Paralelo

### ✓ Selección Exclusiva

Se aplica en el diagrama de las Quejas o Reclamaciones General (Ver Figura Nº 1) donde en el modo de presentación de la queja se ejecuta sólo una rama del proceso: Entregar personalmente o Enviar por correo postal:

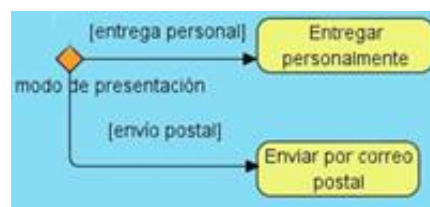


Figura Nº 4 Patrón Selección Exclusiva

Patrones de Sincronización y Enrutamiento Avanzada



# Capítulo 2. Modelado del Negocio

## ✓ Mezcla Múltiple o Serialización

Se ve reflejado en el subproceso Materia Penal (Ver Anexo 12) donde se unen dos ramas del proceso: la de Modificar Informe y la de Mantener el informe en el mismo estado. Al final siendo cual sea la acción que se realice el informe se envía a la dirección de PDC.

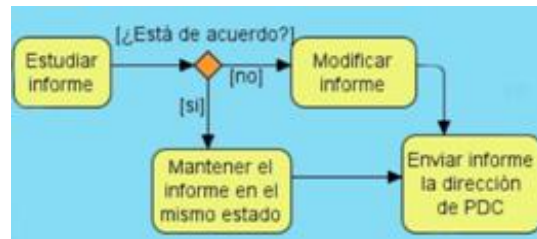


Figura Nº 5 Patrón Mezcla Múltiple o Serialización

### 2.2.3 Descripción de subprocesos de negocio.

La descripción de los procesos de negocio permite describir en qué consiste cada una de las actividades que se realizan en el proceso. A continuación se muestra el DPN de uno de los subprocesos en cuestión y su descripción a modo de ejemplificación. Las demás descripciones se encuentran en el paquete descripciones de procesos pertenecientes al expediente del proyecto.

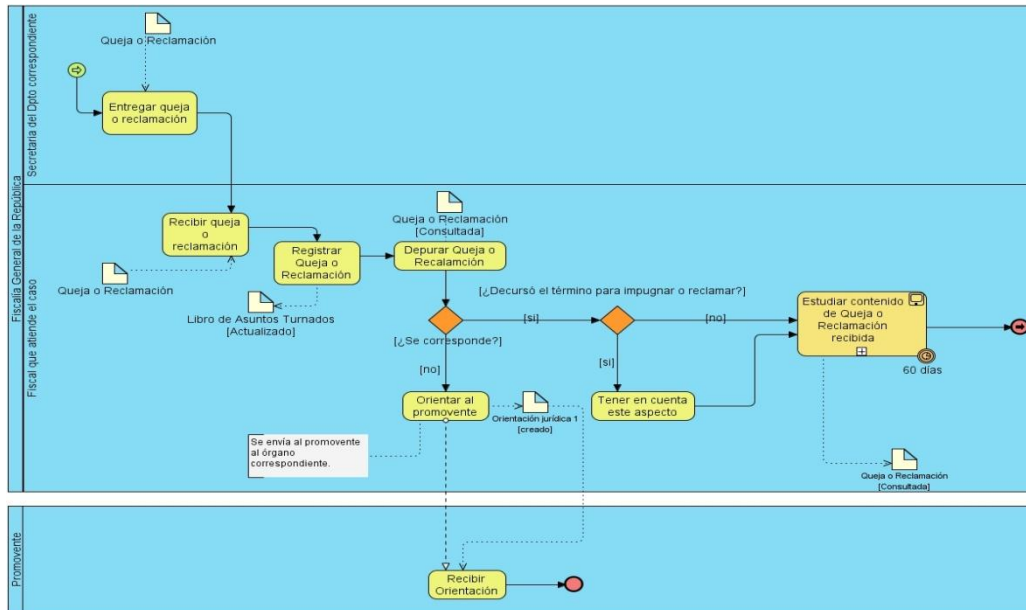


Figura Nº 6 Diagrama de Procesos de Negocio del subproceso Asuntos administrativos y laborales.



## Capítulo 2. Modelado del Negocio

Tabla 3 Descripción de procesos de negocio del sub-proceso Asuntos administrativos y laborales.

<b>Objetivo</b>	Investigar en el orden legal las quejas y reclamaciones de la población ante presuntas violaciones de sus derechos en materia administrativa o laboral.
<b>Evento(s) que lo genera(n)</b>	Se recibe una queja o reclamación por parte de una persona natural.
<b>Pre condiciones</b>	Ha ocurrido presunta violación de derechos.
<b>Marco legal</b>	Ley 83/99, Reglamento de la Ley 83/99 e Instrucciones del Fiscal General.
<b>Clientes internos</b>	N/A
<b>Clientes externos</b>	Promovente.
<b>Entradas</b>	Queja o Reclamación.
<b>Flujo de eventos</b>	
<b>Flujo básico</b> del sub-proceso Asuntos administrativos y laborales.	
1.	<b>Entregar queja o reclamación:</b> La secretaria del Dpto. que corresponde entrega la queja o reclamación al fiscal que tramitará el caso.
2.	<b>Recibir queja o reclamación:</b> El fiscal recibe la queja o reclamación.
3.	<b>Registrar Queja o Reclamación:</b> El fiscal registra la queja o reclamación en el libro de asuntos turnados.
4.	<b>Depurar Queja o Reclamación:</b> El fiscal se lee la queja o reclamación, la consulta para ver de qué se trata.
5.	Si el fiscal decide que no se corresponde con una queja o reclamación que debe ser tramitada por él, ir al flujo alterno 5.a de lo contrario. Si el fiscal comprueba que decursó el término para impugnar o reclamar ir al flujo alterno 5.b de lo contrario. <b>Estudiar contenido de Queja o Reclamación recibida:</b> Ir a la descripción del subproceso Estudiar contenido de Queja o Reclamación recibida.
<b>Pos-condiciones</b>	
1.	Promover una Queja o Reclamación en materia administrativa o laboral.
2.	Actualizar el libro de asuntos turnados.
<b>Salidas</b>	



## Capítulo 2. Modelado del Negocio

1.	Libro de asuntos turnados.
2.	Respuesta a la Queja o Reclamación. (DOC).
<b>Flujos paralelos</b>	
N/A	
<b>Pos-condiciones</b>	
N/A	
<b>Salidas</b>	
N/A	
<b>Flujos alternos</b>	
<b>5. a:</b> Si el fiscal decide que no se corresponde con una queja o reclamación que debe ser tramitada por él.	
1. <b>Orientar al promovente:</b> Se le da una orientación al promovente para que se dirija al órgano correspondiente para tramitar esa queja o reclamación.	
2. <b>Recibir Orientación:</b> El promovente recibe la orientación y termina el proceso de queja o reclamación.	
<b>Pos-condiciones</b>	
1. Se crea la orientación jurídica.	
<b>Salidas</b>	
Orientación Jurídica 1.	
<b>5. b:</b> Si el fiscal comprueba que decursó el término para impugnar o reclamar.	
1. <b>Tener en cuenta este aspecto:</b> El fiscal tiene en cuenta que decursó el término a la hora de pronunciarse.	
2. Continuar con el flujo básico 5.	
<b>Pos-condiciones</b>	
N/A	
<b>Salidas</b>	
N/A	
<b>Asuntos pendientes</b>	
N/A	

Es necesario aclarar que, en la FGR las quejas y las reclamaciones reciben el mismo tratamiento, es decir, no hay diferencia entre una y otra en cuanto a tramitación y respuesta.



## Capítulo 2. Modelado del Negocio

---

### 2.2.4 Modelo Conceptual

Ésta es una herramienta muy importante donde no se define ninguna operación sino que se muestran conceptos, relaciones entre ellos y atributos de los mismos. Se representa en UML mediante un grupo de estructura estática (Diagramas de Clases Conceptuales). Se realizó el Modelo Conceptual (Ver Anexo 15) correspondiente al proceso Quejas o Reclamaciones de la población en la fase de Modelado del Negocio y se hizo un refinamiento del mismo en la fase de Especificación de Requisitos.

### 2.2.5 Validación del modelado de negocio del proceso Quejas o Reclamaciones

La validación de los artefactos generados en esta fase fue posible a través de las Revisiones Técnicas Formales (RTF) y la validación con el cliente. A continuación se muestran los detalles de estas dos formas de validación:

#### *RTF*

Ésta es una actividad de garantía de los sistemas informáticos, son reuniones del personal técnico con el objetivo de descubrir problemas de calidad en el producto a revisar. Su aplicación a la documentación técnica permite detectar deficiencias, ambigüedades, omisiones, errores, tanto de formato como de contenido.

Se realizaron RTF al modelado procesos de negocio por parte de la dirección del proyecto: Analista Principal, Líder de Proyecto y Jefe de Módulo. Gracias a esto fue posible detectar errores y omisiones que habían sido cometidas en el modelado que luego fueron corregidos. Una vez concluidos los artefactos a general en esta fase fueron revisados por el equipo de trabajo y aprobados por los fiscales implicados. Con esta técnica se logra ofrecer a los clientes mayor seguridad ya que se le garantiza que los artefactos cuentan con la calidad requerida desde el punto de vista técnico.

#### *Validación con el cliente*

Esta forma de validación fue importante para lograr la satisfacción del cliente. Se realizaron varios encuentros con los fiscales que atienden las quejas o reclamaciones en las diferentes materias y se analizaron posibles errores que, en cuanto a términos fiscales habían sido mal empleados. Tras una serie de cambios y modificaciones en este sentido, el modelado de los procesos de negocio fue finalmente aceptado por los clientes obteniendo la carta de aceptación de los mismos. (Ver Anexos 16, 17, 18, 19 y 20).





## *Capítulo 2. Modelado del Negocio*

---

### **2.3 Conclusiones parciales**

Con la realización de las actividades correspondientes a esta fase, se logró modelar todo el negocio que encierra el Proceso Quejas o Reclamaciones. De esta forma se generaron los artefactos correspondientes: Descripción de Proceso de Negocio, Mapas de Procesos y el Modelo Conceptual los cuales servirán de base para las próximas fases.



## Capítulo 3. Especificación de Requisitos

---

### CAPÍTULO 3. ESPECIFICACIÓN DE REQUISITOS

#### 3.1 Introducción

Este capítulo está dedicado a la **Especificación de Requisitos** teniendo como objetivo principal lograr un punto de convergencia entre analistas y clientes de modo que queden identificadas de forma correcta las funcionalidades que debe tener el sistema. Los artefactos generados en el Modelado del Negocio fueron el punto de partida para generar los entregables correspondientes a esta disciplina, los cuales son: Especificación de Requisitos de Software y Especificación de Casos de Uso. Los mismos serán validados a partir de los prototipos y las métricas correspondientes.

#### 3.2 Fases de la disciplina

Como parte de la Ingeniería de Requisitos se realizaron las actividades correspondientes a cada una de las fases que propone el PM en esta disciplina. A continuación se irán mostrando los resultados obtenidos a medida que se vaya analizando cada fase:

##### 3.2.1 Obtención de requisitos

Para llevar a cabo esta actividad se aplicó la combinación de las técnicas definidas en el primer capítulo con el objetivo de lograr mejores resultados en este sentido. Se obtuvo un total de 50 RNF y 54 RF. Los RF resultaron ser la base para que una vez agrupados, resultara un total de 29 Casos de Uso (CU).

##### Requisitos Funcionales obtenidos

Teniendo en cuenta el concepto de RF planteado en la fundamentación teórica del trabajo, se identificaron los siguientes:

RF\_PDC\_QR\_ 1: Registrar queja o reclamación.

RF\_PDC\_QR\_ 2: Buscar queja o reclamación.

RF\_PDC\_QR\_ 3: Consultar queja o reclamación.

RF\_PDC\_QR\_ 4: Consultar libro de control.

RF\_PDC\_QR\_ 5: Crear boleta de despacho.

RF\_PDC\_QR\_ 6: Consultar boleta de despacho.



## Capítulo 3. Especificación de Requisitos

---

RF\_PDC\_QR\_ 7: Crear boleta de traslado.

RF\_PDC\_QR\_ 8: Consultar boleta de traslado.

RF\_PDC\_QR\_ 9: Consultar Libro de Registro de Entrada de PDC.

RF\_PDC\_QR\_ 10: Registrar entrada de queja o reclamación a PDC.

RF\_PDC\_QR\_ 11: Asignar queja o reclamación a Dpto. correspondiente.

RF\_PDC\_QR\_ 12: Consultar Libro de Registro de Entrada al Dpto. correspondiente.

RF\_PDC\_QR\_ 13: Registrar entrada de queja o reclamación al Dpto. correspondiente.

RF\_PDC\_QR\_ 14: Buscar antecedentes de la queja o reclamación.

RF\_PDC\_QR\_ 15: Adjuntar antecedentes.

RF\_PDC\_QR\_ 16: Crear Acuse de recibo.

RF\_PDC\_QR\_ 17: Consultar Libro de Registro de Salida de PDC.

RF\_PDC\_QR\_ 18: Registrar queja o reclamación en el libro de salida de PDC.

RF\_PDC\_QR\_ 19: Crear Expediente pre-judicial.

RF\_PDC\_QR\_ 20: Consultar Expediente pre-judicial.

RF\_PDC\_QR\_ 21: Modificar Expediente pre-judicial.

RF\_PDC\_QR\_ 22: Crear dictamen.

RF\_PDC\_QR\_ 23: Consultar Dictamen.

RF\_PDC\_QR\_ 24: Modificar Dictamen.

RF\_PDC\_QR\_ 25: Crear registro de notas.

RF\_PDC\_QR\_ 26: Modificar registro de notas.

RF\_PDC\_QR\_ 27: Crear acta de cierre de expediente.

RF\_PDC\_QR\_ 28: Crear informe del caso.

RF\_PDC\_QR\_ 29: Consultar informe del caso.



## Capítulo 3. Especificación de Requisitos

---

- RF\_PDC\_QR\_30: Modificar informe del caso.
- RF\_PDC\_QR\_31: Crear expediente de queja o reclamación.
- RF\_PDC\_QR\_32: Consultar expediente de queja o reclamación.
- RF\_PDC\_QR\_33: Actualizar expediente de queja o reclamación.
- RF\_PDC\_QR\_34: Crear Acta de comparecencia.
- RF\_PDC\_QR\_35: Consultar Acta de comparecencia.
- RF\_PDC\_QR\_36: Modificar Acta comparecencia.
- RF\_PDC\_QR\_37: Crear Orientación jurídica 1 y 2.
- RF\_PDC\_QR\_38: Consultar Orientación jurídica 1 y 2.
- RF\_PDC\_QR\_39: Modificar Orientación jurídica 1 y 2.
- RF\_PDC\_QR\_40: Crear Citación oficial.
- RF\_PDC\_QR\_41: Crear Resolución.
- RF\_PDC\_QR\_42: Consultar Resolución.
- RF\_PDC\_QR\_43: Modificar Resolución.
- RF\_PDC\_QR\_44: Registrar prórroga.
- RF\_PDC\_QR\_45: Consultar prórroga.
- RF\_PDC\_QR\_46: Crear escrito de solución de prórroga.
- RF\_PDC\_QR\_47: Crear escrito de comunicación.
- RF\_PDC\_QR\_48: Crear boleta de notificación.
- RF\_PDC\_QR\_49: Modificar boleta de notificación.
- RF\_PDC\_QR\_50: Crear Pronunciamiento no violatorio.
- RF\_PDC\_QR\_51: Consultar Pronunciamiento no violatorio.
- RF\_PDC\_QR\_52: Modificar Pronunciamiento no violatorio.



## Capítulo 3. Especificación de Requisitos

---

RF\_PDC\_QR\_ 53: Crear libro de asuntos turnados.

RF\_PDC\_QR\_ 54: Actualizar libro de asuntos turnados.

### Requisitos No Funcionales obtenidos:

Los RNF están enfocados a propiedades o cualidades que debe tener el software. Partiendo de aquí, se definieron los siguientes:

#### Usabilidad

- ✓ El software tendrá siempre la posibilidad de ayuda disponible para cualquier tipo de usuario, lo que le permitirá un avance considerable en la explotación de la aplicación en todas sus funcionalidades.
- ✓ Existirán servidores locales con capacidad necesaria para el procesamiento de las solicitudes del conjunto de aplicaciones de las diferentes oficinas.
- ✓ Las aplicaciones siempre solicitarán los datos a través del servidor local.
- ✓ Desde cada servidor local se establecerá la conexión con servidores centrales para mantener la actualización de los datos en ambos sentidos.
- ✓ El tiempo de entrenamiento requerido para que usuarios normales y avanzados sean productivos operando el sistema es de 15 días.
- ✓ Debe poseer una interfaz agradable para el cliente.

#### Fiabilidad

- ✓ El sistema estará disponible 24 horas al día, 7 días a la semana.
- ✓ Disponibilidad de los casos asignados desde cualquier parte del país.
- ✓ Tiempo medio entre fallos es de 1 mes.
- ✓ El tiempo permitido para que el sistema quede fuera de operación luego de haber fallado es de 2 días.
- ✓ La precisión y exactitud requerida en las salidas del sistema o sea el máximo de errores, es de 5



## Capítulo 3. Especificación de Requisitos

---

errores/MLC.

- ✓ La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores.

### Eficiencia

- ✓ Tiempo de respuesta promedio de las peticiones que se realizan al servidor no deberá ser mayor de 3 segundos.
- ✓ El número de clientes o transacciones que el sistema puede alojar es de 2000.

### Soporte

- ✓ Soporte para grandes volúmenes de datos y velocidad de procesamiento.
- ✓ Tiempo de respuesta rápido en accesos concurrentes.
- ✓ El sistema debe ser multiplataforma.

### Restricciones de diseño

- ✓ El lenguaje de programación es PHP 5.2.0.
- ✓ El *framework* de desarrollo es *Symfony* 1.3.6.
- ✓ La herramienta IDE de desarrollo a utilizar será Eclipse 3.5+PDT.
- ✓ La herramienta CASE utilizada es el *Visual Paradigm* 6.0.
- ✓ La herramienta gestor de base de datos es el *Postgre SQL* 8.4.

### Interfaz

- ✓ El sistema tiene que ofrecer una interfaz amigable, fácil de operar.
- ✓ Diseño sencillo, con pocas entradas, permitiendo que no sea necesario mucho entrenamiento para que los usuarios puedan utilizar el sistema.
- ✓ El sistema tiene que mantener la línea de diseño establecida para la institución que mantiene la uniformidad y representatividad de la misma.

### Requerimientos de Seguridad



## Capítulo 3. Especificación de Requisitos

---

- ✓ Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.
- ✓ El sistema debe mantener en todo momento la seguridad de la información asegurando la autenticidad de la misma.
- ✓ La seguridad se establecerá por roles que se le asignarán a los usuarios que interactúen con el sistema.
- ✓ El software brindará solamente aquellas funcionalidades que competen a la Unidad Ejecutora donde esté implantado.
- ✓ El sistema mantendrá en todo momento las trazas que se corresponden con las diferentes situaciones críticas que se puedan ocurrir.

### Hardware

#### *Cliente*

- ✓ Memoria RAM 256 MB o superior.
- ✓ Disco duro de 20 GB o superior.
- ✓ Procesador Intel® a 1 GHz de velocidad de procesamiento o superior.
- ✓ Tarjeta de red Ethernet.
- ✓ Debe tener acceso a una impresora.
- ✓ Debe tener acceso a un scanner.

#### *Servidores*

- ✓ Se debe contar con un servidor que brinde las siguientes funciones integradas: DNS, DHCP, Firewall.
- ✓ Se debe contar con un servidor de correo independiente.
- ✓ Se debe contar con un servidor independiente LDAP.
- ✓ Se debe contar con un servidor *Postgre SQL* y FTP.
- ✓ Se debe contar con un servidor independiente de Salva.



## Capítulo 3. Especificación de Requisitos

---

- ✓ Quad-Core Intel® Xeon® E5430 (2.66 GHz, 80 Watts, 1333 FSB) o superior.
- ✓ Memoria RAM de 32 GB o superior

### Software

#### *Cliente*

- ✓ Navegador Firefox 3 o superior

#### *Servidor*

- ✓ Sistema Operativo *Debian GNU/Linux 5*
- ✓ *Postgre SQL 8.4*
- ✓ PHP 5.3 con las librerías y configuraciones necesarias desplegar *Symfony 1.3*

### Requisitos Legales, de Derecho de Autor y otros.

El Sistema de Gestión Fiscal debe garantizar el cumplimiento de lo dispuesto en las normas jurídicas, fundamentalmente las que se emitieron con rango de Ley, permitiendo adecuar el sistema de forma fácil, a cambios en dichas normas. Debe igualmente hacer referencia al Registro y depósito legal de la aplicación.

### Estándares Aplicables

El estándar utilizado en el proyecto para la especificación de requisitos es el IEEE 830. Los estándares de codificación utilizados en el proyecto están basados en normas seguidas por el código *Symfony*, específicamente el estándar “*Upper Camel Case*” para nombre de clases y variables.

#### **3.2.2 Análisis de requisitos**

Una vez culminada la etapa de obtención de requisitos en la que se identificaron los RF y RNF mencionados anteriormente, se llevó a cabo de una manera muy profunda el análisis de estos requisitos con la intención de detectar requisitos mal identificados, inconsistencias y ambigüedades en los mismos.

#### **3.2.3 Especificación de requisitos**

En esta fase se generó el artefacto correspondiente que propone el PM donde se especifican por separado todos los requisitos. Para ello se hizo uso de la técnica plantillas o patrones con el objetivo de describir los requisitos de una forma centralizada en la plantilla correspondiente definida por el PM. A





## Capítulo 3. Especificación de Requisitos

continuación se cita un ejemplo muy puntual de especificación de requisitos Crear Acta de Comparecencia con el fin de dar a entender en qué consiste. Las demás especificaciones se encuentran en el documento Especificación de Requisitos del Software perteneciente al expediente de proyecto.

**Tabla 4 Especificación de Requisitos para el requisito Crear Acta de Comparecencia**

Crear acta de comparecencia	Crear el acta de comparecencia	A
-----------------------------	--------------------------------	---

**Prototipo**

FISCALÍA GENERAL DE LA REPÚBLICA  
DIRECCIÓN DE PROTECCIÓN DE LOS DERECHOS CIUDADANOS

Fiscalía  de

Siendo las  del día  comparece ante el fiscal que suscribe

El ciudadano(a)  vecino de  
 con carné de identidad No.  
 teléfono  quien labora en el centro  
 ocupando el cargo de  
 con teléfono  quien manifiesta lo siguiente:

Siendo todo, firman la presente en la fecha consignada.

Compareciente Fiscal actuante



## Capítulo 3. Especificación de Requisitos

Campo	Tipo de Dato	Regla o Restricción
<ul style="list-style-type: none"><li>Fiscalía</li></ul>	<ul style="list-style-type: none"><li>String</li></ul>	Se selecciona si la fiscalía es provincial o municipal.
<ul style="list-style-type: none"><li>De</li></ul>	<ul style="list-style-type: none"><li>String</li></ul>	Se selecciona el municipio o provincia a la que pertenece la fiscalía
<ul style="list-style-type: none"><li>Hora</li></ul>	<ul style="list-style-type: none"><li>Date</li></ul>	N/A
<ul style="list-style-type: none"><li>Día</li></ul>	<ul style="list-style-type: none"><li>Date</li></ul>	N/A
<ul style="list-style-type: none"><li>Nombre del fiscal</li></ul>	<ul style="list-style-type: none"><li>String</li></ul>	Campo que sólo admite letras.
<ul style="list-style-type: none"><li>Ciudadano (a)</li></ul>	<ul style="list-style-type: none"><li>String</li></ul>	Campo que sólo admite letras.
<ul style="list-style-type: none"><li>Dirección ciudadano</li></ul>	<ul style="list-style-type: none"><li>String</li></ul>	Campo que admite letras, números y símbolos.
<ul style="list-style-type: none"><li>Número de carnet</li></ul>	<ul style="list-style-type: none"><li>Integer</li></ul>	Campo que sólo admite números.
<ul style="list-style-type: none"><li>Número de teléfono</li></ul>	<ul style="list-style-type: none"><li>Integer</li></ul>	Campo que sólo admite números.
<ul style="list-style-type: none"><li>Centro de trabajo</li></ul>	<ul style="list-style-type: none"><li>String</li></ul>	Campo que admite letras y números.
<ul style="list-style-type: none"><li>Manifiesto del cliente</li></ul>	<ul style="list-style-type: none"><li>String</li></ul>	Campo que admite letras y números
<b>Observaciones</b>	N/A	

Como se había mencionado anteriormente, de la agrupación de estos RF se identificaron los CU del sistema, esto constituye una alternativa para representar las funcionalidades del sistema partiendo de las acciones que realizará el usuario. A continuación se muestra el diagrama de casos de uso correspondiente al proceso Quejas o Reclamaciones:



# Capítulo 3. Especificación de Requisitos

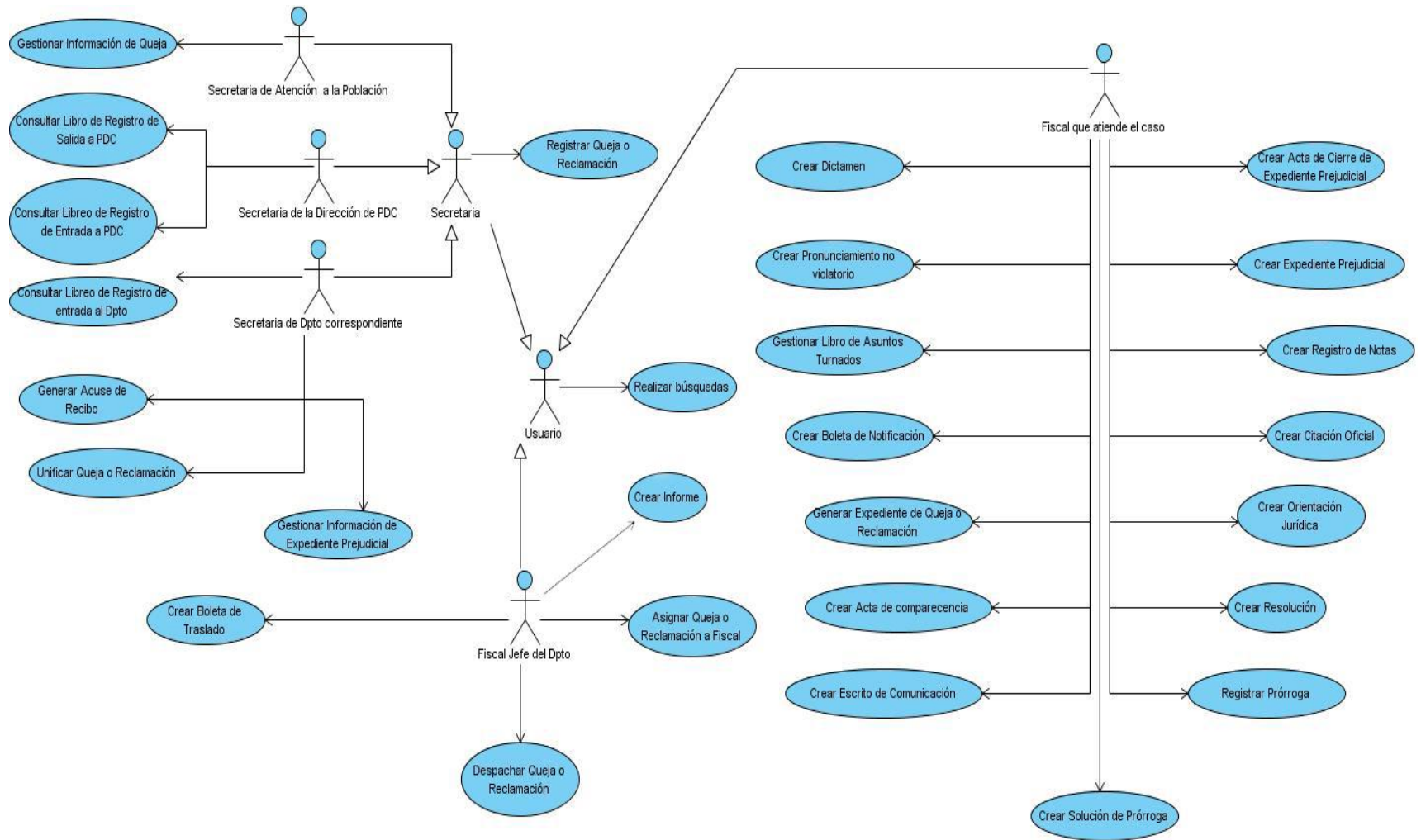


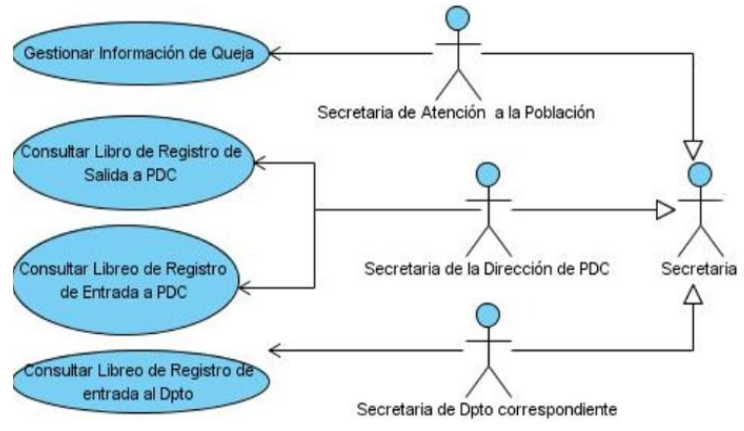
Figura Nº 7 Diagrama de Casos de Uso del Sistema.



# Capítulo 3. Especificación de Requisitos

## Aplicación de los patrones

El patrón Roles Comunes se ve evidenciado en el Diagrama de Casos de Uso del sistema como se muestra en la figura:



**Figura Nº 8 Patrón Roles Comunes**

## Especificación de casos de uso

Se realiza con el objetivo de describir cada una de las acciones que realiza el sistema tras las opciones que selecciona el usuario. Fue utilizada la técnica plantillas o patrones para su elaboración. A continuación se muestra la especificación del CU Registrar queja o reclamación. Las demás especificaciones se encuentran en el documento Especificación de casos de uso perteneciente al expediente de proyecto.

**Tabla 5 Especificación de casos de uso para el CU Registrar Queja o Reclamación.**

<b>Objetivo</b>	Crear un escrito para comunicarse con alguna persona requerida según sea necesario.
<b>Actores</b>	Fiscal.
<b>Resumen</b>	El caso de uso inicia cuando el actor accede a la opción registrar queja o reclamación y el sistema muestra una interfaz con los datos necesarios. El caso de uso termina cuando se ha registrado la queja o reclamación.
<b>Complejidad</b>	Alta.
<b>Prioridad</b>	Crítico.
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. El sistema debe estar correctamente instalado.</li> <li>2. El actor debe estar autenticado con los permisos necesarios.</li> </ol>
<b>Postcondiciones</b>	Se ha registrado una queja o reclamación.
<b>Flujo de eventos</b>	



# Capítulo 3. Especificación de Requisitos

Flujo básico Registrar queja o reclamación.		
	Actor	Sistema
1.	Accede a la opción Registrar queja o reclamación.	1.1 El sistema muestra la interfaz correspondiente solicitando los siguientes datos. ✓ No. Queja o Reclamación ✓ Fecha ✓ Promovente Y las opciones ✓ Guardar ✓ Cancelar
2	Llena los datos y selecciona la opción Guardar.	2.1 Verifica que los datos introducidos sean correctos 2.2 Registra los datos 2.3 Muestra un mensaje indicando que sus datos han sido guardados correctamente.
3		Termina el caso de uso


**Prototipo de interfaz de usuario**

Sus datos se han guardado correctamente.

## Flujos alternos



## Capítulo 3. Especificación de Requisitos

2. a: El actor selecciona la opción Cancelar.			
	<b>Actor</b>		<b>Sistema</b>
1	Selecciona la opción Cancelar.		Muestra la interfaz principal.
2.b: Los datos introducidos no son correctos			
1.			1.1 Muestra un mensaje de error en la parte superior indicándole al usuario: Los datos introducidos no son correctos. 1.2 Volver al paso 2 del flujo básico.
 Los datos introducidos no son correctos. Por favor, corrija los siguientes errores y vuelva a guardar sus datos:			
<b>Sección 1: "Nombre"</b>			
Flujo básico			
	<b>Actor</b>		<b>Sistema</b>
1.			
<b>Flujos alternos</b>			
Nº Evento			
	<b>Actor</b>		<b>Sistema</b>
1.			
<b>Relaciones</b>	CU Incluidos	N/A	
	CU Extendidos	N/A	
<b>Requisitos no funcionales</b>			
<b>Asuntos pendientes</b>	N/A		

### 3.2.4 Validación de requisitos del proceso Quejas o Reclamaciones

Como parte de esta fase dentro de la ingeniería de requisitos fue realizada la validación de los requisitos identificados a través de las siguientes técnicas:

*RTF*

Se llevaron a cabo por parte de la dirección del proyecto al igual que en el Modelado del Negocio. Esto facilitó la detección de inconsistencias en los requisitos, ambigüedades y hasta omisiones que habían sido cometidas en el proceso de obtención de los requisitos. Una vez generados todos los artefactos fueron nuevamente revisados por el equipo de proyecto y por los clientes implicados hasta obtener el resultado esperado.

*Validación con el cliente*



## Capítulo 3. Especificación de Requisitos

---

Esta técnica de validación tuvo lugar a través de los prototipos de interfaz de usuario dándole de esta manera una visión más palpable al cliente de las funcionalidades del sistema. Se revisaron cada uno de los prototipos con los fiscales correspondientes por materias y aunque se detectaron inconsistencias en algunos de los casos, los errores fueron corregidos. Se adicionaron campos en algunos prototipos tras una reunión realizada con el Fiscal General en el centro. Todos estos asuntos fueron analizados y corregidos en su momento, quedando aprobados finalmente por el cliente.

### *Aplicación de la métrica Estabilidad en los Requisitos*

Es necesario asegurarse de que los requisitos sean lo más estables posibles antes de pasar a la próxima fase del proyecto. Para esto se aplicó la métrica Estabilidad en los Requisitos la cual se calcula de la siguiente manera:

$$ETR = [(RT - RM) / RT] * 100$$

Donde:

ETR: valor de la estabilidad de los requisitos.

RT: total de requisitos definidos.

RM: número de requisitos modificados, que se obtienen como la sumatoria de los requisitos insertados, modificados y eliminados.

Teniendo en cuenta que se identificaron un total de 54RF, dentro de los cuales 18 resultaron modificados (RM), se calcula:

$$ETR = [(54 - 18) / 54] * 100 = 66.6$$

Del resultado obtenido luego de aplicar esta métrica, se deduce que los requisitos no fueron lo suficientemente estables debido a cambios que se realizaron al comienzo del desarrollo de la fase. Una vez corregidos todos los señalamientos o sugerencias se realizaron nuevamente las validaciones en conjunto con los clientes implicados y los resultados fueron finalmente satisfactorios.

### Métrica para evaluar los casos de uso

Para evaluar la calidad de los casos de uso se utilizó un modelo de métricas orientado a objetos el cual, como se explicaba en el primer capítulo, tiene en cuenta cuatro atributos: Completitud, Correctitud,



## Capítulo 3. Especificación de Requisitos

Consistencia y Complejidad. Estos atributos se analizan en términos de un conjunto de factores, cada uno de los cuales tendrá asociada una métrica. En la siguiente tabla se muestran los resultados obtenidos:

**Tabla 6 Resultados de la métrica para evaluar los casos de uso**

<b>Atributo</b>	<b>Factor</b>	<b>Métrica asociada</b>	<b>Valor</b>
<b>Complejidad</b>	<b>Factor 1.</b> ¿Han sido definidos todos los roles relevantes de usuario encargados de generar/modificar o consultar información?	<b>Métrica 1:</b> Número de roles relevantes omitidos.	Número de roles relevantes omitidos: 0  <b>Resultado: 100%</b>
	<b>Factor 2.</b> ¿Se presenta una descripción resumida de todos los casos de uso?	<b>Métrica 2:</b> Número de casos de uso que no tienen descripción resumida.	Número de casos de uso que no tienen descripción resumida: 0 <b>Resultado: 100%</b>
	<b>Factor 3:</b> ¿Están definidos todos los requisitos que justifican la funcionalidad del caso de uso?	<b>Métrica 3:</b> Número de requisitos omitidos por caso de uso.	Número de requisitos omitidos por caso de uso: 0 <b>Resultado: 100%</b>
	<b>Factor 4:</b> Todos los casos de uso han sido clasificados de acuerdo a su relevancia: (Crítico, secundario, auxiliar y opcional).	<b>Métrica 5:</b> Número de casos de uso que no han sido clasificados.	Número de casos de uso que no han sido clasificados: 0 <b>Resultado: 100%</b>
			Se presenta un 100%.
<b>Consistencia</b>	<b>Factor 5.</b> ¿El nombre dado a los casos de uso es una expresión verbal que describe alguna	<b>Métrica 6:</b> Número de casos de uso que tienen un nombre incorrecto.	Número de casos de uso que tienen un nombre incorrecto: 0 <b>Resultado: 100%</b>





## Capítulo 3. Especificación de Requisitos

	funcionalidad relevante en el contexto del usuario?		
	<b>Factor 6.</b> ¿Está adecuadamente redactado (en el lenguaje del usuario) el flujo de eventos?	<b>Métrica 7:</b> Grado de adecuación de la descripción del flujo de eventos para un caso de uso. <b>Métrica 8:</b> Número de casos de uso no aceptados.	La descripción se define en el lenguaje del usuario. Se define el responsable de cada acción. <b>Resultado: 100%.</b>
	<b>Factor 7.</b> ¿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?	<b>Métrica 9:</b> Número de casos de uso cuya descripción extendida no inicia con una acción externa o con una condición monitoreada por el sistema.	Número de casos de uso cuya descripción extendida no inicia con una acción externa o con una condición monitoreada por el sistema: 0 <b>Resultado: 100%.</b>
			<b>Se presenta un 100%.</b>
<b>Correctitud</b>	<b>Factor 9.</b> ¿Representa el caso de uso requisitos comprensibles por el usuario?	<b>Métrica 10:</b> Número de casos de uso en que los requisitos representados no son comprensibles por el usuario.	Número de casos de uso en que los requisitos representados no son comprensibles por el usuario: 0 <b>Resultado: 100%.</b>
	<b>Factor 10.</b> ¿Existe para cada caso de uso de negocio por lo menos un usuario responsable?	<b>Métrica 11:</b> Número de casos de uso que no tienen un usuario responsable.	Número de casos de uso que no tienen un usuario responsable: 0 <b>Resultado: 100%.</b>



## Capítulo 3. Especificación de Requisitos

	<b>Factor 11.</b> ¿Se ajusta la representación del diagrama de casos de uso de acuerdo a lo normado en la metodología?	<b>Métrica 12:</b> Grado en el que se ajusta el diagrama de casos de uso a la metodología.	Grado en el que se ajusta el diagrama de casos de uso a la metodología. <b>Resultado: 100%</b>
	<b>Factor 12.</b> ¿Las interacciones definidas introducen mejoras al proceso actual?	<b>Métrica 13:</b> Número de casos de uso que deben ser modificados para mejorar el proceso actual.	Número de casos de uso que deben ser modificados para mejorar el proceso actual: 1 <b>Resultado: 94.74%.</b>
			<b>Se presenta un 98.69%</b>
<b>Complejidad</b>	<b>Factor 13.</b> ¿Los elementos dentro del diagrama están adecuadamente ubicados de manera que facilitan su interpretación?	<b>Métrica 14:</b> Número de elementos del diagrama que requieren reubicación.	Número de elementos del diagrama que requieren reubicación: 0 <b>Resultado: 100%</b>
			<b><u>Resultado: 100%.</u></b>

Al aplicar estas métricas se pudo evaluar los factores completitud, consistencia, correctitud y complejidad del diagrama de casos de uso del sistema. Los resultados arrojados dieron la posibilidad de comprobar que los requisitos que se identificaron son tratados al menos en un caso de uso, con lo que se demuestra que todas las necesidades expresadas por el cliente están presentes. También se verificó que los casos de uso fueron descritos de forma clara, mostrando el flujo alterno separado del flujo básico lo que da una mayor legibilidad de los mismos. Todo lo mostrado anteriormente permite evidenciar que el artefacto caso de uso del sistema se encuentra con la calidad requerida para seguir con los flujos posteriores del desarrollo del software.



## Capítulo 3. Especificación de Requisitos

---

### **3.2.5 Administración de requisitos**

Esta fase tuvo lugar de forma paralela con las anteriores ya que constituye una buena práctica a aplicar durante esta disciplina y durante todo el proceso de desarrollo.

### **3.3 Conclusiones parciales**

Luego de haber realizado las actividades correspondientes a esta disciplina se logró entender y realizar una buena identificación de requisitos, llevar un estricto control de los mismos identificando sus inconsistencias y ambigüedades. Se generaron de manera eficiente los artefactos necesarios para continuar con el desarrollo de la siguiente fase del proyecto: Especificación de Requisitos de Software y Descripción de Requisitos de Software.



### CAPÍTULO IV. DISEÑO

#### 4.1 Introducción

La confección de este capítulo está basada en la realización del diseño de los procesos referentes a Quejas o Reclamaciones de la población elaborando los diagramas de clase y de paquete correspondientes. Se hace uso de los artefactos generados hasta el momento para obtener los correspondientes a esta disciplina, en este caso se trata del Modelo de Diseño. La validación tendrá lugar con la aplicación de las métricas TC y APH.

#### 4.2 Diseño

El diseño es uno de los pilares fundamentales en la elaboración del software. El diseño propuesto tiene que cumplir a cabalidad con los requerimientos del sistema ya que es la única forma de convertir exactamente los requisitos de un cliente en un producto o sistema de software finalizado... (38). Debe ser capaz de facilitar las mejoras del software, tiene que ser entendible por otros profesionales de la especialidad de manera que permita la comprobación del sistema fácilmente.

##### Framework de desarrollo seleccionado

Se seleccionó *symfony* como *framework* de desarrollo por estar diseñado para optimizar el desarrollo de las aplicaciones web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web mediante el uso del patrón Modelo-Vista-Controlador (MVC):

En este patrón el Modelo administra el comportamiento y los datos del dominio de aplicación. Mientras que la Vista transforma el modelo en una página web manejando la visualización de la información con la que el usuario interactúa. El Controlador, por su parte, es quien interpreta las acciones del ratón y el teclado, informando al modelo y/o a la vista para que cambien según resulte apropiado. Este es el patrón con el cual está implementado el *framework* de desarrollo a utilizar. A continuación se muestra la explicación concreta de cómo *symfony* implementa cada uno de los patrones de diseño ya explicados conceptualmente en el capítulo 1:

##### Patrones de Diseño que implementa *symfony* (39)

##### Patrones Grasp

- ✓ *Creador*: En la clase *Actions* se encuentran todas las acciones definidas para el módulo en cuestión y se ejecutan cada una de ellas. En las acciones se crean los objetos de las clases que



representan las entidades, evidenciando de este modo que la clase *Actions* es "creador" de dichas entidades.

- ✓ *Experto*: Es uno de los más utilizados, puesto que Propel es la librería externa que utiliza Symfony para realizar su capa de abstracción en el modelo, encapsula toda la lógica de los datos y son generadas las clases con todas las funcionalidades comunes de las entidades.
- ✓ *Alta Cohesión*: *Symfony* permite asignar responsabilidades con una alta cohesión, por ejemplo la clase *Actions* tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las propiedades, es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios.
- ✓ *Controlador*: Todas las peticiones web son manejadas por un solo controlador frontal (*sfActions*), que es el punto de entrada único de toda la aplicación en un entorno determinado. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con la URL entrada por el usuario.
- ✓ *Bajo Acoplamiento*. La clase *Action* hereda solamente de *sfActions* para lograr un bajo acoplamiento de clases.

### Patrones GOF

*En la categoría Estructurales:*

*Decorator (Envoltorio):*

- ✓ Añade funcionalidad a una clase, dinámicamente. El archivo *layout.php*, que también se denomina plantilla global, almacena el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el *layout*, o si se mira desde el otro punto de vista, el *layout* decora la plantilla. Este comportamiento es una implementación de dicho patrón de diseño.
- ✓ *Composite (Objeto compuesto)*: Permite tratar objetos compuestos como si de uno simple se tratase. Sirve para construir objetos complejos a partir de otros más simples y similares entre sí, gracias a la composición recursiva y a una estructura en forma de árbol. Esto simplifica el tratamiento de los objetos creados, ya que al poseer todos ellos una interfaz común, se tratan todos de la misma manera.

*En la categoría Creacionales:*

- ✓ *Singleton (Instancia única)*: Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. En el controlador frontal hay una llamada a `sfContext::getInstance()`. En una acción, el método `getContext()`, un objeto muy útil que guarda una referencia a todos los objetos del núcleo de *Symfony*.
- ✓ *Abstract Factory (Fábrica abstracta)*: Permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí y haciendo transparente el tipo de familia concreta que se esté usando. Cuando el *framework* necesita por ejemplo crear un nuevo objeto para una petición, busca en la definición de la factoría el nombre de la clase que se debe utilizar para esta tarea.

#### 4.2.1 Diagrama de paquetes del diseño

Los diagramas de paquetes son útiles para organizar y manipular la complejidad subsistemas grandes ya que permite dividir los mismos en partes pequeñas fáciles de manejar. Este tipo de diagrama tiene su repercusión en los diagramas de clase del diseño agrupando las clases dentro del paquete que le corresponda. Para lograr una mejor organización y especialización de los elementos del diseño se definieron los siguientes paquetes del diseño, siguiendo el patrón arquitectónico MVC, por ser el patrón que utiliza el *framework symfony*. Los paquetes pueden tener relación con otros paquetes para mostrar las dependencias entre ellos:

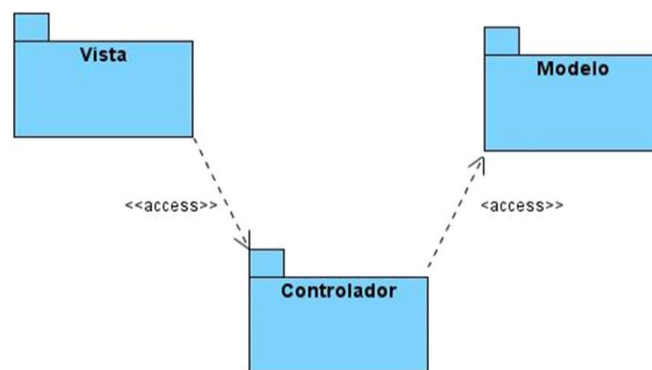






Figura Nº 9 Diagrama de Paquetes

#### 4.2.2 Diagrama de clases del diseño

Los diagramas de clases del diseño “describen” la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Es en estos diagramas donde se crea el diseño conceptual de la información y los componentes que manejará en el sistema.

Para la confección de los diagramas de clases se utilizó el *Visual Paradigm* como herramienta CASE y se realizó el modelado con UML. De la misma forma para la representación de las clases y sus funcionalidades, se hizo uso de varios elementos significativos:

**Tabla 7 Estereotipos web**

Estereotipos web correspondientes a las clases	
Estereotipo	Descripción
 <p><b>Client Page</b></p>	Una instancia de página cliente es una página web, con formato HTML.
 <p><b>Form</b></p>	Colección de elementos de entrada que son parte de una página cliente. Tiene una correspondencia directa con la etiqueta <FORM> de XHTML.
 <p><b>Server Page</b></p>	Representa la página web que tiene script, que se ejecuta en el servidor. Estos scripts interactúan con los recursos que se encuentran al alcance del servidor. Sólo puede mantener relaciones con objetos que se encuentren en el servidor.
 <p><b>ClientScriptObject</b></p>	Es una colección de scripts del lado del cliente que existe como un archivo separado y que son incluidos mediante una petición independiente por parte del navegador.

A continuación se muestra un ejemplo de los diagramas de clases del diseño, en este caso representa el diseño del CU Gestionar Acta de Comparecencia. El diseño correspondiente al resto de los casos de uso se encuentra en el documento Modelo de diseño perteneciente al expediente de proyecto.

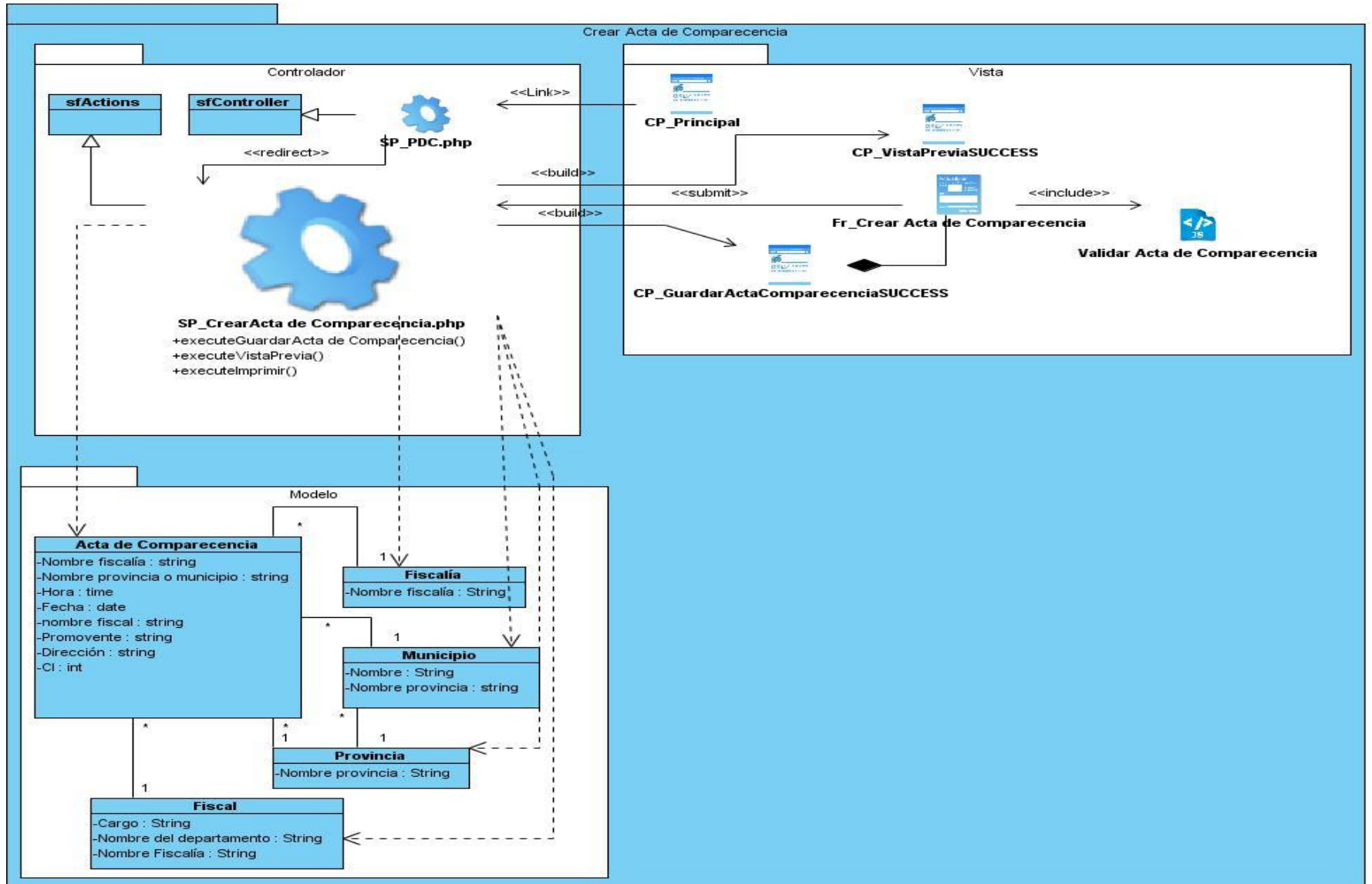


Figura Nº 10 Diagrama de Clases del Diseño.



✓ Descripción de clases

A continuación se muestra una breve descripción de las clases que componen el diagrama anterior. Las demás descripciones de clases se encuentran en el documento Modelo del diseño perteneciente al expediente de proyecto.

**Nombre:** Clase SP\_CrearActadeComparecencia.php.

**Propósito:** Controlar la creación de un Acta de Comparecencia.

**Tipo de clase:** Controladora

**Descripción:**



**SP\_CrearActadeComparecencia.php**

```
+executeGuardarActa_de_Comparecencia()
+executeVistaPrevia()
+executeImprimir()
```

**Tabla 8 Descripción de las clases del diseño.**

Nombre del Método	Descripción
Execute_GuardarActa_de_Comparecencia ()	Permite Guardar un Acta de Comparecencia una vez creada.
Execute_Imprimir ()	Convierte el documento a formato PDF para imprimir.
Execute_VistaPrevia ()	Muestra una vista de cómo quedará el documento creado.

4.2.3 Validación del diseño del proceso Quejas o Reclamaciones

La importancia del diseño del software se puede describir con una sola palabra –calidad–. (38). La calidad en el diseño es el grado con que las especificaciones de diseño (especificaciones acerca de los distintos componentes del producto) responden a las expectativas del consumidor. (40). Por estos motivos es importante que el producto tenga la calidad requerida ya que de aquí dependerá del nivel de calidad obtenido en el diseño del mismo.

Cuando se mide la calidad de un producto de software es importante aplicar métricas que arrojen resultados útiles para la medición. El objetivo de aplicar métricas en este sentido es, sencillamente

mejorar pues esto ayuda a la identificación de problemas y a descubrir oportunidades para enriquecer la calidad del sistema y el rendimiento del mismo.

A continuación se muestran los resultados obtenidos tras la aplicación de la métrica tamaño de clase:

✓ Tamaño de clase (TC).

**Tabla 9 Representación del tamaño de las clases del diseño.**

No.	Clases	Cantidad de atributos	Cantidad de operaciones	Tamaño
1	SP_Despachar Queja o Reclamación	1	3	Pequeño
2	Boleta de despacho	18	2	Pequeño
3	Provincia	1	3	Pequeño
4	Queja o reclamación	1	3	Pequeño
5	Promovente	1	3	Pequeño
6	Fiscal	3	3	Pequeño
7	SP_Buscar Boleta de Notificación	1	4	Pequeño
8	Boleta de Notificación	8	3	Pequeño
9	SP_Crear Solución de prórroga	1	3	Pequeño
10	Solución de Prórroga	10	2	Pequeño
11	Tipo de Solución	3	2	Pequeño
12	SP_Registrar Prórroga	1	3	Pequeño
13	Prórroga	3	2	Pequeño
14	SP_Buscar Dictamen	1	3	Pequeño
15	Dictamen	6	3	Pequeño
16	SP_Buscar Expediente Prejudicial.php	1	3	Pequeño
20	Expediente prejudicial	7	2	Pequeño
21	SP_Buscar Resolución	1	3	Pequeño
22	Resolución	12	3	Pequeño
23	Fiscalía	1	3	Pequeño
24	Municipio	2	3	Pequeño
25	SP_Buscar Registro de Notas	1	3	Pequeño
26	Registro de notas	4	3	Pequeño
27	SP_Buscar Pronunciamiento no Violatorio	1	3	Pequeño
28	Pronunciamiento no violatorio	10	3	Pequeño
29	SP_Buscar Orientación Jurídica	1	3	Pequeño
30	Orientación jurídica	10	3	Pequeño
31	SP_Buscar Informe Respuesta	1	3	Pequeño



32	Informe	10	3	Pequeño
33	SP_Buscar Escrito	1	3	Pequeño
34	Escrito de Comunicación	10	2	Pequeño
35	SP_Buscar Acta de Comparecencia	1	3	Pequeño
36	Acta de Comparecencia	8	3	Pequeño
37	SP_Registrar Queja o Reclamación	1	3	Pequeño
38	SP_Gestionar Libro de Asuntos Turnados	1	3	Pequeño
39	Libro de Asuntos turnados	8	3	Pequeño
40	SP_Crear Pronunciamiento no Violatorio	1	3	Pequeño
41	SP_Crear Boleta de Notificación	1	3	Pequeño
42	SP_Crear Resolución	1	3	Pequeño
43	SP_Crea Orientación Jurídica	1	3	Pequeño
44	SP_Crear Acta de Comparecencia	1	3	Pequeño
45	SP_Crear Informe	1	3	Pequeño
46	SP_Crear Registro de Notas	1	3	Pequeño
47	SP_Crear Escrito Comunicación	1	3	Pequeño
48	SP_Crear Citación Oficial	1	3	Pequeño
49	SP_Crear Acta Cierre Expediente Prejudicial	1	3	Pequeño
50	SP_Crear Dictamen	1	3	Pequeño
51	SP_Crear Expediente Prejudicial.	1	3	Pequeño
52	SP_Crear Acuse de Recibo.	1	3	Pequeño
53	Acuse de recibo	5	2	Pequeño
54	SP_Crear Boleta de Traslado Bajo Control	1	3	Pequeño
55	Boleta de traslado bajo control	6	2	Pequeño
56	SP_Crear Boleta de Traslado Definitivo	1	3	Pequeño
57	Boleta de traslado definitivo	9	2	Pequeño
58	SP_Despachar Queja o Reclamación	1	3	Pequeño

Se trabajó con un total de 55 clases para un promedio de 3.4 atributos por clases y 2.85 de operaciones como a continuación se muestra:

**Tabla 10 Resultados de la Métrica Tamaño de Clases**

Cantidad de clases	Promedio de atributos	Promedio de operaciones
55	3,4	2,85



Luego de haber aplicado esta métrica, se obtuvo como resultado que las clases del diseño son pequeñas en su totalidad debido a que las 58 clases que fueron analizadas todas tienen tamaño pequeño. Esto demuestra que las mismas no tienen grandes responsabilidades por lo que permite la reutilización de clases demostrando que el sistema tendrá una fácil implementación. Por tanto se puede afirmar que los resultados obtenidos por la métrica son positivos.

### ✓ Árbol de Profundidad de Herencia (APH)

Esta métrica se define como la longitud máxima desde el nodo hasta la raíz del árbol. A medida que crece el APH, es más probable que las clases de niveles inferiores hereden muchos métodos. Esto da lugar a posibles dificultades cuando se intenta predecir el comportamiento de una clase. Una jerarquía de clases profunda con un valor grande de APH, lleva también a una mayor complejidad del diseño. Por el lado positivo, los valores grandes de APH implican que se pueden reutilizar muchos métodos. Por su parte, algunos autores como Lorenz y Kidd sugieren que un umbral de 6 niveles como indicador es un abuso en la herencia en distintos lenguajes de programación.

A partir de los datos obtenidos al aplicar la métrica APH se obtuvo que el nivel de profundidad de herencia más alto entre las clases del diseño es dos, por lo que queda demostrado que el diseño no es complejo, ya que existe un bajo acoplamiento entre las clases y no es difícil su mantenimiento, pues el nivel de profundidad de la herencia es el mínimo que puede existir en la relación entre clases, este resultado conlleva a que el diseño realizado tiene calidad.

### **4.3 Conclusiones Parciales**

Al realizar el diseño de los procesos referentes a las quejas o reclamaciones de la población se tiene una visión más profunda de lo que va a ser el sistema. Se obtuvo el Modelo de diseño correspondiente con sus respectivos diagramas de clases y de paquetes logrando una estructura más centralizada de las clases del diseño. Se aplicó la métrica de TC y la de APH, ambas arrojaron resultados satisfactorios y con la calidad requerida.

## **CONCLUSIONES GENERALES**

Luego de realizar el presente trabajo se puede concluir que:

- ✓ El estudio y análisis del PM permitió seleccionar herramientas y lenguajes que logran ajustarse a las necesidades del proceso Quejas y Reclamaciones.
- ✓ Fueron utilizadas varias técnicas y patrones durante las fases desarrolladas que permitieron obtener resultados fiables los cuales se vieron reflejados en cada uno de los artefactos generados.
- ✓ Las técnicas y métricas utilizadas para validar los entregables obtenidos en cada una de las disciplinas, propiciaron la correcta medición de los mismos garantizando así resultados eficientes que logran contribuir con la pronta implementación del sistema.

## **RECOMENDACIONES**

Para complementar los resultados obtenidos en este trabajo, se recomienda:

1. Realizar talleres con la dirección del proyecto, analistas que no participaron en el proceso y desarrolladores de modo que los procesos sean entendibles por todo el equipo de desarrollo, garantizando así la obtención de un buen producto de software.
2. Mantener el estudio del PM con el fin de mantener actualizado el trabajo e insertar el mismo en el nivel 2 de CMMI.
3. Realizar estricto seguimiento de los requisitos identificados durante las próximas fases e iteraciones que propone el ciclo de vida para proyectos incluido en el PM, con el objetivo de obtener un producto que realmente satisfaga las expectativas.
4. Mantener latente una activa comunicación con los clientes e ir refinando al mismo tiempo, las recomendaciones o sugerencias que se obtengan del trabajo en equipo.
5. Realizar la implementación correspondiente a los procesos referentes a Quejas o Reclamaciones de la población.

## BIBLIOGRAFÍA

1. *Ley de la Fiscalía General de la República.*
2. *Indicaciones metodológicas para la atención a la población.*
3. Pereira, Lic. Julio A.Fernández. *Derecho Procesal Penal. Tema I.* Universidad de La Habana : s.n.
4. *Ley de Procedimiento Civil, Administrativo, Laboral y Económico.*
5. *Decreto Ley 99/87. Artículo 1.*
6. La Rama Judicial de Puerto Rico. [Online] <http://www.ramajudicial.pr/orientacion/imp-quejas.html>.
7. Complaints Board. [Online] <http://www.complaintsboard.com/?search=quejas>.
8. Sommerville, Ian. *Ingeniería del Software.*
9. *Metodologías Ágiles o formales y robustas.* Pérez, Dr.C Profesor Auxiliar Pedro Y. Piñero Pérez y Dr.C Profesor Auxiliar Pedro Y. Piñero.
10. Calisoft. [Online] <http://calisoft.uci.cu/index.php/proceso-de-mejora>.
11. *Programa de Mejora. Introducción a la Administración de Requisitos. Tema I.* Calisoft.
12. Ingeniero Software. [Online] <http://www.ingenierosoftware.com/calidad/cmm-cmmi.php>.
13. Machado, Ing. Yenier Figueroa. *Proyecto Técnico del Sistema de Informatización de la Gestión de las Fiscalías II.* 2011.
14. *CMMI-DEV, v1.2 CMU/SEI-2006-TR-008. IEEE 610.12.*
15. Davis. 1993.
16. Robertson, Suzanne y Robertson, James. *Mastering the Requirements Process.* s.l. : Second Edition, 2006.
17. Sommerville., Ian. *Ingeniería Del Software.* . s.l. : Sexta edición.
18. Kotonya, G y Sommerville. *I. Requirements Engineering: Processes and Techniques.* 2000.
19. — . *I. Requirements Engineering: Processes and Techniques.* 2000.

20. M.J. Escalona, N. Koch. *Ingeniería de Requisitos en Aplicaciones para la Web: Un estudio comparativo*. España : s.n.
21. Overgaard, Gunnar y Palmkvist, Karin. *Use Cases Patterns and Blueprints*. 2004.
22. Stephen., A. White. *Process Modeling Notation and workflow Patterns*. .
23. M.J. Escalona, N. Koch. *Ingeniería de Requisitos en Aplicaciones para la Web: Un estudio comparativo*. España : s.n., 2010.
24. Ingenieros de Software. [Online] <http://www.ingenierossoftware.com/analisisydiseno/patrones-diseno.php>.
25. LARMAN C. *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. México: PRENTICE HALL : s.n., 1999.
26. Larman. 1999.
27. Gamma. 1995.
28. Bernárdez. 2004.
29. S. R Chidamber, C. F Kemerer. *IEEE Transactions on Software Engineering*. "A metric suite forsz object oriented design". 1994.
30. Kendall. *Análisis y Diseño de Sistemas*. 1997.
31. *Conferencia 3. Herramientas para la Administración de Requisitos. Programa de Mejora*.
32. Visual Paradigm. [Online] <http://www.visual-paradigm.com/aboutus/10reasons.jsp>.
33. Frías, Iliana de la Rosa Zayas. *Análisis del módulo Despacho de Pasajeros, en la Aduana General de la República*. 2010.
34. otros, Jacobson y. 2000.
35. Pressman. *Aprendiendo UML en 24 horas*.
36. Aprendiendo UML en 24 horas. [Online] <http://www.scribd.com/doc/3708746/Aprendiendo-UML-en-24-horas>.
37. Buenas tareas. [Online] <http://www.buenastareas.com/ensayos/Arquitectura-Del-Software/922875.html> .





38. Pressman. 2005.
39. Symfony users. [Online] <http://groups.google.com/group/symfony-users/msg/cd94d2ddb2057355>.
40. Berlinches. 2004.
41. Catálogo único de trámites y servicios. [Online]  
[http://www.tramitesyservicios.df.gob.mx/index.php?option=com\\_content&view=article&id=406:quejas-en-el-funcionamiento-del-transporte-publico-bicitaxis-y-atencion-a-problemas-viales&catid=104:quejas-y-demandas-vecinales&Itemid=484](http://www.tramitesyservicios.df.gob.mx/index.php?option=com_content&view=article&id=406:quejas-en-el-funcionamiento-del-transporte-publico-bicitaxis-y-atencion-a-problemas-viales&catid=104:quejas-y-demandas-vecinales&Itemid=484).
42. Reynoso, Carlos Billy. *Introducción a la Arquitectura de Software*. [Online] Marzo 2004 .  
<http://www.willydev.net/descargas/prev/IntroArq.pdf>.
43. INTERCOM, Grupo. 1997-2010.
44. *Área de Proceso de Medición y Análisis de CMMI-DEV v 1.2*. Calisoft. Marzo 2010.
45. *Documento de Arquitectura de Software SIGEF I*.
46. Revista Vinculando. [Online] 2003-2011.  
[http://vinculando.org/articulos/sociedad\\_america\\_latina/propuesta\\_guia\\_de\\_medidas\\_para\\_evaluacion\\_sistemas\\_informacion.html](http://vinculando.org/articulos/sociedad_america_latina/propuesta_guia_de_medidas_para_evaluacion_sistemas_informacion.html) .
47. Roger S. Pressman. *Ingeniería de Software. Un enfoque práctico*.

## GLOSARIO DE TÉRMINOS

### **Términos informáticos:**

**BPMN:** *Business Process Modeling Notation*. En su traducción al español Notación de Modelado de Procesos de Negocio.

**CASE:** *Computer Aided Software Engineering*. En su traducción al español significa Ingeniería de Software Asistida por Computación.

**Framework:** Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Un *framework* puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

**GRASP:** *General Responsibility Assignment Software Patterns*. En su traducción al español Patrones de Software de Asignación de Responsabilidades.

**IEEE:** Son las siglas en inglés de *The Institute of Electrical and Electronics Engineers* (El Instituto de Ingenieros Eléctricos y Electrónicos), una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas.

**Tecnologías de la información y la comunicaciones (TIC):** Conjunto de avances tecnológicos que proporcionan la informática, las telecomunicaciones y las tecnologías audiovisuales, que comprenden los desarrollos relacionados con los ordenadores, Internet, la telefonía, las aplicaciones multimedia y la realidad virtual. Estas tecnologías básicamente proporcionan información, herramientas para su proceso y canales de comunicación.

**Software:** Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora.