

**Universidad de las Ciencias Informáticas.  
Facultad 3**



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

**Título:** Ingeniería de Requisitos del procedimiento  
Diligencias Previas al proceso ejecutivo del Módulo  
Económico del Proyecto Tribunales Populares Cubanos  
(TPC).

**Autor:** Angel Alexander Guillén Suárez.

**Tutor:** Ing. Yurisleidys Leiva Zuñiga.

**Asesor:** Ing. Chavelys Téllez Larramendi.

**La Habana, Cuba.  
Junio 2011**

# Declaración de Autoría

---

## DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 3 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

---

Angel Alexander Guillén Suárez

**Autor**

---

Ing. Yurisleidys Leiva Zuñiga

**Tutor**

---

Ing. Chavelys Téllez Larramendi

**Asesor**

**" ... Si los jóvenes fallan, todo fallará. Es mi más profunda convicción que la juventud cubana luchará por impedirlo. Creo en ustedes."**

**Fidel Castro Ruz.**



**" ... La revolución no se lleva en los labios para vivir de ella, se lleva en el corazón para morir por ella. "**

**Ernesto Che Guevara.**

## Agradecimientos

A toda mi familia, por el apoyo que me han dado en toda mi vida, por estar siempre ahí, por su preocupación, por sentirse orgullosos de mí, en especial a mis abuelos Clara y Lenzo, todas mis tías y tíos, a los primos en especial a Juan Alberto, Guillermito y Samil, muchas gracias a todos ustedes, los quiero.

A mi amigo de siempre Michel, por ser incondicional, por ayudarnos, por pelear y al momento seguir como si nada hubiese pasado, por aconsejarme cuando me hizo falta, por nunca olvidarse de mí, gracias a ti Miche.

A todas las personas del barrio que siempre me ayudaron, se preocupaban, que al menos un día me dijeron, ¿dime Alex como va la escuela esa?, en especial a Kirenia, Ñica, Cecilia, Iliana, Yannelis, a mis niñas: Gretchel y Gretel, a todos ustedes, gracias.

A mis amigos de mis primeros años, Carlos Abel y Damián, por los buenos momentos que pasamos siempre, por los estudios, las fiestas, nunca los olvidare.

A Maydel y Lilibel, por siempre aconsejarme a decirles las mejores cosas a las niñas, por los buenos consejos que yo les di también, por quererme tanto y por siempre ayudarnos cuando lo necesitábamos, las quiero un mundo.

A todos los del aula, Janio, Yasma, Perdo, Erneston, el Pollo, Gilbe, Elizabeth, Yudiel, Abilio, Daneysi, Daylen, Mariem, Adneris, Gretel, Gustavo, Ismail, Leo, Asdrubal, algunos se fueron de la escuela pero al menos los ayudamos, los demás llegamos hasta el final. Felicidades y éxitos.

A los del grupo de Venezuela, por los buenos y malos momento que pasamos, también por trabajar y ayudarnos en las tierras lejanas cumpliendo con el deber, a todos ustedes muchas gracias.

A mis mejores socios, esos que me dicen "Papa", por ser yo el mas viejo del piquete, esos que siempre me han aguataado en todo momento y que seguirán, con ustedes la he pasado perfecto, confió en sus títulos de ingeniero, Alfredo, Yandi, Jiubel, Sergio y Ramón.

A Arielito, como me distes trabajo hijo mío, gracias a tu familia también, eres una persona excelente, te quiero, recuerdo cuando te fuiste, que días aquellos, pero al final prometiste venir para mi tesis y aquí estás, nunca te olvidaré.

A Juanjo y Arturo, por tampoco olvidarse de mí, por los momentos de cuero, por joder tanto, los quiero mis hermas.

# Agradecimientos

---

A Pavel, Adrian Fonth, David, Alain, el Rafa, Ismael, Toni, Pepe a todo esa pila de colegas de 3ro, me disculpan si me queda alguien, se darán cuenta que hacer esto no es muy fácil, lo importante es que siempre podrán contar conmigo.

A julio por estar siempre ahí cuando lo necesite, por los consejos, las cosas buenas y las malas, voy a ti también, cuídate mi herma, eres el próximo ingeniero.

A Soraya, por tanta ayuda, por los grandes consejos, por tener tanta paciencia conmigo, por ponerme a recitar el ppt, muchas gracias mi ángel, te quiero, eres otra madre para mi, te quiero.

A Linnet, por toda las cosas que ha hecho por mi, gracias por todo, siempre te tendré presente, cuídate mucho que personas que ayudan ya no quedan muchas, un abrazo grande.

A Yurita, por ser una persona especial, por las muchas cosas que hiciste y que harás, todavía te queda buena faena, nunca te olvidaré. Te quiero.

A todos los del proyecto Aduana que me ayudaron en algún momento, a los del Proyecto Tribunales, a los profes que tuve a lo largo de la vida de estudiante, a todo el que se preocupo, muchas gracias.

A Chavelys por su apoyo, por tener tanta paciencia, por comprenderme siempre, gracias a ti.

A el Tribunal por hacerme estudiar tanto y prepararme, gracia a ustedes aprendí mucho.

A las personas que me ayudaron en Venezuela, a toda la brigada, con ustedes pase momentos inolvidables, conocí personas incondicionales, Kati, Arelis, Rodo, Amarilis, a todos ustedes muchas gracias.

Gracias a todos aquellos que me han ayudado de alguna forma a lo largo de mi vida, aquí tienen un amigo, nunca dejen de luchar, siempre se puede y siempre se aprende, un abrazo grande.

## Dedicatoria

A mi madre, por ser mi guía en todo momento, por inculcarme siempre buenos valores, por siempre estar ahí cuando lo he necesitado, a ti mamita te dedico esta tesis. TE QUIERO MUCHO MUCHO.

A mi hermana, por siempre estar ahí dándome esos buenos consejos, por ser siempre el ejemplo a seguir, ahora nos queda Marquito, ese será el próximo profesional, TE QUIERO MUCHO.

A mi padre, por apoyarme y estar siempre ahí cuando lo he necesitado. Te quiero.

A Marquitos, por la confianza en su futuro, por lo especial que es para mí, TE QUIERO MUCHO BEBE.

## RESUMEN

Los Tribunales Populares Cubanos (TPC) son una entidad con el objetivo fundamental de aplicar lo que corresponde a los litigios contractuales que le sean de su interés. En la Universidad de las Ciencias Informáticas (UCI) se está desarrollando una solución informática con el objetivo de informatizar los procedimientos que se realizan en los TPC, dicho solución está conformada por 7 módulos o subsistemas siendo uno de ellos el Módulo Económico perteneciente a la sala de lo Económico de dichos Tribunales.

En el presente trabajo se realizó un estudio del estado del arte de los sistemas que actualmente gestionan el proceso judicial en Cuba y el mundo. También se investigó acerca de las metodologías de desarrollo de software, los lenguajes de modelado así como las notaciones para describir los procesos que se realizan en la sala de lo Económico, además de las herramientas CASE (Computer Aided Software Engineering) con el objetivo de seleccionar la más conveniente para el desarrollo del procedimiento Diligencias Previas perteneciente al Módulo Económico del TPC.

Se realizó la ingeniería de requisitos del procedimiento Diligencias Previas obteniéndose los requerimientos y el modelo de Casos de Uso del Sistema, así como la aplicación de métricas con el objetivo de validar los artefactos generados anteriormente y comprobar que cumplieran con las especificidades del cliente.

**Palabras claves:** Tribunales Populares Cubanos, ingeniería de requisitos, procedimiento Diligencias Previas del Módulo Económico.

## ÍNDICE

<b>INTRODUCCIÓN.....</b>	<b>1</b>
<b>CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.....</b>	<b>4</b>
1.1 INTRODUCCIÓN.....	4
1.2 SOLUCIONES INFORMÁTICAS QUE GESTIONAN EL PROCESO JUDICIAL.....	4
1.2.1 Soluciones Informáticas en el Mundo.....	4
1.2.2 Soluciones Informáticas en Cuba.....	5
1.3 NECESIDAD DE SOLUCIÓN INFORMÁTICA PARA LA GESTIÓN DEL PROCESO JUDICIAL.....	5
1.4 METODOLOGÍAS DE DESARROLLO DE SOFTWARE.....	5
1.4.1 XP (Extreme Programming).....	6
1.4.2 Proceso Unificado Racional (RUP).....	7
1.4.3 Metodología Scrum.....	8
1.4.4 Justificación de la Metodología Seleccionada.....	9
1.5 INGENIERÍA DE REQUISITOS.....	10
1.6 TÉCNICAS PARA LA CAPTURA DE REQUISITOS.....	11
1.7 LENGUAJES Y NOTACIONES PARA LA MODELACIÓN DE PROCESOS.....	14
1.7.1 Cadena de proceso guiada por eventos.....	15
1.7.2 Lenguaje de Modelado Unificado (UML).....	15
1.7.3 Notación para la Modelación de Procesos de Negocio (BPMN en sus siglas en inglés).....	16
1.8 HERRAMIENTAS PARA EL MODELADO.....	16
1.8.1 Rational Rose.....	17
1.8.2 Visual Paradigm.....	18
1.8.3 Enterprise Architect.....	19
1.8.4 Justificación de la Herramienta de Modelado Seleccionada.....	19
1.9 HERRAMIENTAS PARA MODELADO DE PROTOTIPOS NO FUNCIONALES.....	20
1.9.1 Microsoft Office Visio 2007.....	20
1.9.2 BPWin 4.1.....	20
1.9.3 Axure RP Pro.....	21
1.9.4 Justificación de la Herramienta para el Modelado de Prototipos no Funcionales seleccionada.....	22
1.10 PATRONES DE CASOS DE USO.....	22

1.11	METRICAS PARA REQUISITOS Y CASOS DE USO .....	24
1.12	CONCLUSIONES PARCIALES.....	25
<b>CAPÍTULO 2: SOLUCIÓN PROPUESTA.....</b>		<b>26</b>
2.1	INTRODUCCIÓN.....	26
2.2	DESCRIPCIÓN DEL PROCESO DE NEGOCIO DE DILIGENCIAS PREVIAS DE LA MATERIA ECONÓMICA	26
2.3	REQUISITOS DEL SISTEMA.....	34
2.3.1	Requisitos Funcionales .....	35
2.3.2	Requisitos no Funcionales .....	37
2.4	DEFINICIÓN DE ACTORES DEL SISTEMA.....	39
2.5	DIAGRAMA DE CASO DE USO DEL SISTEMA.....	41
2.5.1	Realización de los Casos de Uso del Sistema.....	42
2.6	CONCLUSIONES PARCIALES .....	48
<b>CAPÍTULO 3: ANÁLISIS DE LOS RESULTADOS .....</b>		<b>49</b>
3.1	INTRODUCCIÓN.....	49
3.2	MÉTRICA PARA LA CALIDAD DE ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE.....	49
3.3	MÉTRICAS PARA LA FUNCIONALIDAD DEL DIAGRAMA DE CASOS DE USO DEL SISTEMA. ....	50
3.4	ACTA DE ACEPTACIÓN DE LOS REQUISITOS .....	55
3.5	CONCLUSIONES PARCIALES .....	56
<b>CONCLUSIONES.....</b>		<b>57</b>
<b>RECOMENDACIONES.....</b>		<b>58</b>
<b>BIBLIOGRAFÍA.....</b>		<b>59</b>
<b>GLOSARIO DE TÉRMINOS .....</b>		<b>62</b>

## ÍNDICE DE FIGURAS

FIGURA 1.1 METODOLOGÍA RUP. ....	8
FIGURA 1.2 REPRESENTACIÓN GRÁFICA DE UN PROCESO EN IDEF0. ....	15
FIGURA 2.1 BREVE DESCRIPCIÓN DEL PROCEDIMIENTO DILIGENCIAS PREVIAS.....	27
FIGURA 2.2 DIAGRAMA DE CASOS DE USO DEL SISTEMA. ....	41

## INTRODUCCIÓN

En la actualidad el uso de las Tecnologías de la Información y las Comunicaciones (TICs), ha incrementado el desarrollo del software y con el surgimiento de numerosos sistemas que han permitido la informatización de los procesos en una empresa.

En la Universidad de las Ciencias Informáticas (UCI) se está desarrollando un sistema para los Tribunales Populares Cubanos con el objetivo de informatizar todos sus procedimientos, el mismo está conformado por 7 módulos: Civil, Administrativo, Laboral, Económico, Penal, Administración & Gobierno y Reportes. Este trabajo abarcará el procedimiento Diligencias Previas del Módulo Económico, encargado de aplicar lo que corresponda en los litigios contractuales sometidos a su conocimiento, la legislación económica sustantiva, así como cualquier aspecto de carácter procesal que resulte pertinente.

Para darle solución a estos litigios, se atienden de forma manual un enorme cúmulo de expedientes lo que provoca que se cometan errores como la repetición de resoluciones y de asientos en los libros de la sala así como tachaduras y borraduras a la hora de insertar alguna información proveniente de las partes. Estos expedientes son archivados en estantes, lugar donde la búsqueda se torna engorrosa y se encuentran expuestos al deterioro y en ocasiones a la pérdida.

Para lograr un correcto desarrollo del Módulo Económico, se hace necesario un análisis previo, donde se identifiquen las necesidades del cliente, se llegue a un acuerdo entre los involucrados sobre lo que el sistema debe hacer y se traduzcan dichas necesidades a un lenguaje comprensible por los desarrolladores.

Ante la situación expuesta anteriormente surge el siguiente **problema a resolver**: ¿Cómo lograr un entendimiento común entre clientes y desarrolladores, que facilite el desarrollo del Procedimiento Diligencias Previas del Módulo Económico del Proyecto Tribunales Populares Cubanos?

Se plantea como **objeto de estudio** la Ingeniería de Requisitos.

Para dar solución a la problemática descrita se establece como **objetivo general**: realizar la identificación, análisis, especificación y validación del procedimiento Diligencias Previas para transformar las necesidades del cliente a un lenguaje comprensible por los desarrolladores del Módulo Económico del Proyecto del Proyecto Tribunales Populares Cubanos.

El **campo de acción** se enmarca en la identificación, análisis, especificación y validación de los requisitos de software del Procedimiento Diligencias Previas del Módulo Económico del Proyecto Tribunales Populares Cubanos.

Se trazan como **objetivos específicos**:

- Elaborar el marco teórico de la investigación.
- Desarrollar los artefactos relacionados con la ingeniería de requisitos del Procedimiento Diligencias Previas.
- Validar los artefactos obtenidos.

Esta investigación está basada en la siguiente **Idea a defender**: con la realización de un correcto análisis del Procedimiento Diligencias Previas, se logrará el entendimiento entre clientes y desarrolladores, facilitando la implementación del Módulo Económico del Proyecto Tribunales Populares Cubanos.

Se establecen entonces las siguientes **tareas de la investigación**:

- Estudiar las metodologías y herramientas de modelado para el desarrollo de software.
- Realizar un estudio sobre los patrones de Casos de Uso que pueden ser utilizados.
- Definir la estrategia para la captura de los requisitos del sistema.
- Realizar la modelación del negocio.
- Identificar los requisitos funcionales y no funcionales del sistema.
- Estructurar el modelo de casos de uso del sistema.
- Especificar los casos de uso del sistema.
- Definir el prototipo no funcional del sistema.
- Validar los artefactos generados para el sistema.

Con el propósito de obtener los conocimientos a partir de la teoría precedente, los datos de la práctica y elaborar una estrategia general para enfrentar el problema que se investiga, se utiliza el siguiente Diseño metodológico de la investigación científica.

Se estudiaron y aplicaron los siguientes Métodos:

## **Teóricos.**

- **Histórico-Lógico**: Para analizar la trayectoria y evolución de la metodología de desarrollo de software y demás herramientas que se utilizan durante el trabajo.
- **Analítico-Sintético**: Con el propósito de arribar a conclusiones teóricas conceptuales del fenómeno.
- **Modelación**: Para la modelación del negocio es necesario la elaboración de diagramas, figuras y otros artefactos importantes, pues mediante este método se pueden crear abstracciones con el propósito de explicar la realidad.

## Métodos Empíricos.

- **Observación:** Mediante este método se logró percibir de manera directa los problemas existentes en la sala de lo Económico para la realización del procedimiento Diligencias Previas.
- **Entrevista:** Posibilitó la realización de entrevistas individuales y colectivas al personal involucrado en la captura de requisitos, para lograr obtener información precisa de los problemas existentes actualmente y poder entender mejor el funcionamiento de los subprocesos del procedimiento Diligencias Previas.

El **resultado esperado** que se obtendrá con el desarrollo de esta investigación, es lograr un entendimiento entre clientes y desarrolladores para poder realizar una posterior implementación del procedimiento Diligencias Previas.

La presente tesis está estructurada de la siguiente forma:

En el Capítulo 1 se realiza un estudio sobre los procesos que gestionan el proceso judicial en Cuba y el resto del mundo, además de algunas de las metodologías para el desarrollo de software más usadas en la actualidad. También se hace énfasis en los lenguajes de modelado y notaciones para el desarrollo del software, así como las características de la Ingeniería de Requisitos, las técnicas para la captura de requisitos y los patrones de casos de usos que son importantes para lograr una correcta modelación de estos últimos. Se estudian además las características principales de algunas herramientas para el modelado y finalmente se realiza un estudio de las métricas para validar los requisitos de un software y el diagrama de casos de uso.

En el Capítulo 2 se realiza un análisis previo del procedimiento Diligencias Previas aplicándose alguna de las técnicas para la captura de requisitos estudiadas en el capítulo 1. Luego se presentan los requisitos funcionales y no funcionales del sistema, y por último, se muestra el Modelo de Casos de Uso del Sistema, donde se encuentran las especificaciones de los mismos y los prototipos no funcionales.

En el Capítulo 3 se analizan los resultados generados durante el análisis al procedimiento Diligencias Previas. A los requisitos de software obtenidos, se le aplica una métrica para validar la calidad de especificación a los mismo, y al diagrama de casos de uso del Sistema se le aplica una métrica para medir la calidad de la funcionalidad. Por último el cliente expresa mediante una carta de aceptación la conformidad con los artefactos generados demostrando que los mismos cumplieron sus necesidades y expectativas.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

### 1.1 Introducción

En el presente capítulo se hace un estudio sobre el procedimiento económico que se lleva a cabo en los Tribunales Populares Cubanos en todas sus instancias. Se abordan características de soluciones informáticas en el mundo y en Cuba que gestionan el proceso judicial, así como los principales aspectos y objetivos de la propuesta de solución a realizar. Además se estudia la Ingeniería de Requisitos, destacando sus actividades y las técnicas de obtención de requisitos, y se realiza un estudio de los patrones existentes para el modelado del diagrama de casos de uso del sistema, se analizan las tendencias y tecnologías actuales para el desarrollo de software de forma que se pueda justificar las empleadas.

### 1.2 Soluciones Informáticas que Gestionan el Proceso Judicial

En este epígrafe se muestran algunas soluciones informáticas existentes en el mundo que gestionan el proceso judicial en sus respectivos países.

#### 1.2.1 Soluciones Informáticas en el Mundo

**ATLANTE:** Sistema Informático desarrollado en Canarias que registra todas las diligencias realizadas por las distintas instituciones, remite la información adecuada al siguiente paso, aunque los funcionarios esperan que lleguen los documentos en papel para procesarlos. La versión que está en explotación se basa en Lotus Notes y bases distribuidas interconectadas por mensajería. (1)

**EJ:** Sistema desarrollado en el País Vasco que registra todos los procesos judiciales y emite los modelos de acuerdo al proceso correspondiente. Cuenta con procedimientos automáticos para realizar copias de seguridad en diferentes soportes y tiene facilidades para la auditoría informática. Está desarrollado en lenguaje de programación C utilizando bases de datos Oracle y desplegado sobre sistema operativo UNIX. (2)

**LEXNET:** Sistema que actúa sobre la base de la tecnología de los certificados digitales, permite el uso de la firma digital, en los documentos electrónicos, la presentación de escritos y notificaciones. Es un sistema de intercambio de documentos judiciales en formato electrónico. (3)

## 1.2.2 Soluciones Informáticas en Cuba

**SisProP:** Es un sistema informático realizado en la Universidad Central de las Villas Marta Abreu, fue propuesto para abarcar las instancias Supremo y Provincial en la sala penal de sus tribunales. La aceptación por parte de los clientes no fue la mejor ya que el sistema presenta algunas no conformidades para la realización de los procesos implicados en esta sala, no es multiplataforma, está desarrollado con lenguaje de programación Delphi, como sistema gestor de base de datos el SQL Server y no valida casi ningún dato.

**SisEco:** Sistema creado en la Habana en el año 2002 para el área de la estadística en el procedimiento Económico. En él se insertan los documentos radicados manualmente y las salvadas diariamente se guardan en disquetes. La secretaria de estadística recoge el libro de radicación de escritos (LRE) e inserta los datos en la aplicación y cada cinco años borra la información, quedando solamente asentada en los libros.

## 1.3 Necesidad de Solución Informática para la Gestión del Proceso Judicial

A partir del estudio realizado de las soluciones informáticas que gestionan los procesos judiciales en el mundo y en particular en nuestro país, se demuestra que ningún sistema cumple con las necesidades presentadas por el organismo Tribunales Populares Cubanos. Por lo planteado anteriormente se hace necesaria la creación de un sistema que cumpla con los requisitos especificados por el cliente, que sea seguro, que exista interoperabilidad entre sus datos, multiplataforma, que genere los documentos correspondientes a cada proceso y que además brinde la posibilidad de alertar el vencimiento de término de cada proceso y permita obtener reportes estadísticos en tiempo real.

## 1.4 Metodologías de desarrollo de software

Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, evolutivo, incremental, entre otros). La comparación y/o clasificación de metodologías no es una tarea sencilla debido a la diversidad de propuestas y diferencias en el grado de detalle, información disponible y alcance de cada una de ellas. A grandes rasgos, si se adoptan como criterio las notaciones utilizadas para especificar artefactos producidos en actividades de análisis y diseño, se pueden clasificar las metodologías en dos grupos: Metodologías Estructuradas y Metodologías Orientadas a Objetos.

# Capítulo 1: Fundamentación Teórica

---

Por otra parte, considerando su filosofía de desarrollo, aquellas metodologías con mayor énfasis en la planificación y control del proyecto, en especificación precisa de requisitos y modelado, reciben el apelativo de Metodologías Tradicionales (Metodologías Pesadas, o Peso Pesado). Se centran en la definición detallada de los procesos y tareas a realizar, herramientas a utilizar, y requiere una extensa documentación, pues pretende prever todo de antemano. Este tipo de metodologías son más eficaces y necesarias cuanto mayor es el proyecto que se pretende realizar respecto a tiempo y recursos que son necesarios emplear, donde una gran organización es requerida. Una de las metodologías pesadas más conocidas y utilizadas es la Metodología RUP (Rational Unified Process). (4) (5) (6)

Otras metodologías, denominadas Metodologías Ágiles están más orientadas a la generación de código con ciclos muy cortos de desarrollo, se dirigen a equipos de desarrollo pequeños, hacen especial hincapié en aspectos humanos asociados al trabajo en equipo e involucran activamente al cliente en el proceso. Estas metodologías se encargan de valorar al individuo y las iteraciones del equipo más que a las herramientas o los procesos utilizados. Es mucho más importante desarrollar un producto de software que funcione que escribir mucha documentación. El cliente está en todo momento colaborando en el proyecto. Es más importante la capacidad de respuesta ante un cambio realizado que el seguimiento estricto de un plan. Una de las metodologías de este tipo más conocidas y utilizadas es la Metodología XP (Extreme Programming). (4) (5) (6)

A continuación se abordarán características fundamentales de 3 de las metodologías más usadas actualmente; por una parte, XP y Scrum como principales exponentes de las metodologías ágiles, y por el otro, RUP como una de las más generalizadas y usadas entre las metodologías pesadas.

## 1.4.1 XP (Extreme Programming)

Consiste básicamente en ajustarse a una serie de reglas que se centran en las necesidades del cliente para lograr un producto de buena calidad en poco tiempo. Promueve el trabajo en equipo, preocupándose en todo momento del aprendizaje de los desarrolladores y estableciendo un buen clima de trabajo. Este tipo de método se basa en una retroalimentación continuada entre el cliente y el equipo de desarrollo con una comunicación fluida entre todos los participantes, también busca simplificar las soluciones implementadas. Este tipo de programación es la adecuada para los proyectos con requisitos imprecisos, muy cambiantes y con un riesgo técnico excesivo. El ciclo de vida ideal de XP está compuesto por las siguientes seis fases: exploración, planificación de la entrega, iteraciones, producción, mantenimiento y muerte del proyecto, y tiene definidos los siguientes roles: programador, cliente, encargado de pruebas, encargado de seguimiento, entrenador, consultor y gestor. (7)

## 1.4.2 Proceso Unificado Racional (RUP)

El Proceso Racional Unificado es un proceso de ingeniería del software que proporciona un acercamiento disciplinado a la asignación de tareas y responsabilidades en una organización de desarrollo. Su propósito es asegurar la producción de software de alta calidad que se ajuste a las necesidades de sus usuarios finales con unos costos y calendario predecibles. En definitiva RUP es una metodología de desarrollo de software que intenta integrar todos los aspectos a tener en cuenta durante todo el ciclo de vida del software, con el objetivo de hacer abarcables tanto pequeños como grandes proyectos software. (8) (9) (10)

**Las características principales de RUP son:** (9) (10)

- **Guiado/Manejado por casos de uso:** la razón de ser de un sistema software es servir a usuarios ya sean personas u otros sistemas; un caso de uso es una facilidad que el software debe proveer a sus usuarios. Los casos de uso reemplazan la antigua especificación funcional tradicional y constituyen la guía fundamental establecida para las actividades a realizar durante todo el proceso de desarrollo incluyendo el diseño, la implementación y las pruebas del sistema.
- **Centrado en arquitectura:** la arquitectura involucra los elementos más significativos del sistema y está influenciada entre otros por plataformas software, sistemas operativos, manejadores de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados y requisitos no funcionales. Es como una radiografía del sistema que se desarrolla, lo suficientemente completa como para que todos los implicados en el desarrollo tengan una idea clara de qué es lo que están construyendo.
- **Iterativo e Incremental:** para hacer más manejable un proyecto se recomienda dividirlo en ciclos. Para cada ciclo se establecen fases de referencia, cada una de las cuales debe ser considerada como un mini proyecto cuyo núcleo fundamental está constituido por una o más iteraciones de las actividades principales básicas de cualquier proceso de desarrollo. En concreto RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en los distintas actividades.

**Flujos de trabajo de RUP.** (8)

En RUP se definen nueve flujos de trabajo distintos, separados en dos grupos. Los flujos de trabajo de ingeniería son los siguientes:

# Capítulo 1: Fundamentación Teórica

- Modelado del negocio.
- Requisitos.
- Análisis y diseño.
- Implementación.
- Prueba.
- Despliegue.

Los flujos de trabajo de apoyo son:

- Administración del proyecto.
- Configuración y control de cambios.
- Entorno.

Aunque los nombres de los flujos de trabajo de ingeniería recuerden a las etapas de una metodología en cascada, en RUP como se ha visto las fases son distintas, y estos flujos de trabajo serán visitados una y otra vez a lo largo de todo el proceso.

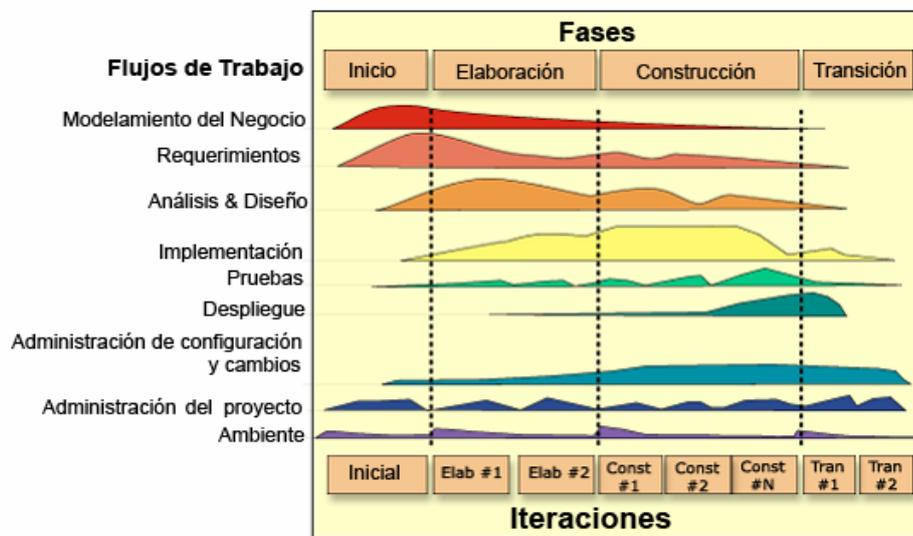


Figura 1.1 Metodología RUP.

## 1.4.3 Metodología Scrum

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa. Se basa en construir primero la

# Capítulo 1: Fundamentación Teórica

---

funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación. (11)

Las actividades definidas en esta metodología se realizan en un periodo de tiempo llamado “sprint”, (la magnitud es definida por el equipo). El conjunto de elementos que forma parte de cada “sprint”, viene dado, por un conjunto de requisitos de alto nivel priorizados, que definen el trabajo a realizar, estos elementos forman parte del “sprint” y son determinados durante la reunión de planificación de “sprint”. Además durante dicha reunión, el cliente del producto identifica los elementos de los requisitos que fueron priorizados para el “sprint”, que quiere ver completados, y los hace del conocimiento del equipo. Entonces el equipo determina la cantidad de ese trabajo que puede comprometerse a completar durante el siguiente “sprint” lo que significa que los requisitos están congelados durante el “sprint”. (11)

Con la aplicación de esta metodología el cliente se compromete con el proyecto dado que lo ve crecer iteración a iteración y le permite en cualquier momento organizar el software y los objetivos de negocio de su empresa, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración y establece sus expectativas indicando el valor que le aporta cada requisito. (11)

## 1.4.4 Justificación de la Metodología Seleccionada

Realizando una valoración de las características de las metodologías de desarrollo expuestas en el presente epígrafe se llega a la conclusión de que se utilizará RUP como metodología de desarrollo de software, ya que es una metodología robusta que se adapta muy bien a proyectos de larga duración, complejos y con un gran equipo de desarrollo como es el caso del proyecto TPC. Por las características del proyecto el cliente no siempre se encuentra presente y es de gran importancia la extensa documentación que ofrece RUP, lo que garantiza poder brindarle al cliente una visión clara de lo que se está realizando en cada etapa del desarrollo.

Además, RUP es una metodología que se encarga de: (12)

- Asegurar la producción de un software de alta calidad que reúna las necesidades de los usuarios finales dentro de un plan y un presupuesto predecible.
- Proveer un enfoque disciplinado para asignar tareas y responsabilidades dentro del desarrollo del sistema.
- Proveer un camino metódico, sistemático para desarrollar, diseñar y validar una arquitectura.
- Reducir en gran medida el riesgo que representa la construcción de sistemas complejos, porque evoluciona de forma incremental partiendo de sistemas más pequeños en los que ya se tiene confianza.

## 1.5 Ingeniería de Requisitos

La Ingeniería de Requisitos en sí cumple un papel primordial en el proceso de construcción y producción de un software, es decir, estará basado en función de las necesidades planteadas por los clientes en un nivel muy general, donde se descubre, documenta, analiza y se definen los servicios o componentes de lo que se desea producir, además de las restricciones que tendrá el producto o software. Su principal tarea consiste en la definición del proceso a seguir en la construcción de un software, y de facilitar la comprensión de lo que el cliente requiera. La obtención correcta de los requisitos puede llegar a describir con claridad, sin ambigüedades, en forma consistente y compacta el comportamiento de un sistema. El principal objetivo es basarse en la extracción de requisitos y sobre todo que sean correctos, que es lo que se pretende en la construcción no solo de grandes sistemas de software sino también en los simples con el propósito de minimizar los problemas relacionados al desarrollo de sistemas en proporción a la realidad de cada proyecto. Con esto se logra reducción de tiempo en la construcción, reducción de errores, y lo más importante no solo para el cliente sino también para el desarrollador, evita gastar dinero más de lo planeado y determinado para el proyecto.

### Actividades de la Ingeniería de Requisitos

La Ingeniería de Requisitos es un proceso que comprende todas las actividades para crear y mantener los requisitos de un sistema, ellas son:

- **Identificación:** esta fase representa el comienzo de cada ciclo y es el nombre comúnmente dado a las actividades involucradas en el descubrimiento de los requisitos del sistema. (7)
- **Análisis:** sobre la base de la identificación realizada previamente, comienza esta fase la cual se enfoca en descubrir problemas con los requisitos del sistema identificados hasta el momento. (7)
- **Especificación:** en esta fase se documentan los requisitos acordados con el cliente, en un nivel apropiado de detalle. (7)
- **Validación:** la validación es la etapa final, su objetivo es ratificar los requisitos, es decir, verificar todos los requisitos que aparecen en el documento especificado para asegurarse que representan una descripción, por lo menos, aceptable del sistema que se debe implementar. Esto implica verificar que los requisitos sean consistentes y que estén completos. (7)
- **Gestión:** es el conjunto de actividades que ayudan al equipo de trabajo a identificar, controlar y seguir los requisitos y sus cambios en cualquier momento, las relaciones entre ellos, las dependencias entre la especificación de requisitos y otros documentos producidos por el

proceso de desarrollo de software. De esta forma se asegura la consistencia entre los requisitos y el sistema construido. Los objetivos principales de la gestión de requisitos son: gestionar la recogida de requisitos, obtener la aprobación de los participantes del proyecto y gestionar los cambios (trazabilidad). (7)

## 1.6 Técnicas para la Captura de Requisitos

La captura 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 (13). El proceso de captura de requisitos puede resultar complejo, principalmente si el entorno de trabajo es desconocido para el equipo de analistas, y depende mucho de las personas que participen en él. Por la complejidad que todo esto puede implicar, la Ingeniería de Requisitos ha trabajado desde hace años en desarrollar técnicas que permitan hacer este proceso de una forma más eficiente y precisa. A continuación se presentan un grupo de técnicas que de forma clásica han sido utilizadas para esta actividad en el proceso de desarrollo de todo tipo de software.

- **Entrevistas:** resulta una técnica muy aceptada dentro de la Ingeniería de Requisitos y su uso está ampliamente extendido. Las entrevistas le permiten al analista tomar conocimiento del problema y comprender los objetivos de la solución buscada. A través de esta técnica el equipo de trabajo se acerca al problema de una forma natural. Existen muchos tipos de entrevistas y son muchos los autores que han trabajado en definir su estructura y dar guías para su correcta realización (14). Básicamente, la estructura de la entrevista abarca tres pasos: identificación de los entrevistados, preparación de la entrevista, realización de la entrevista y documentación de los resultados (protocolo de la entrevista). A pesar de que las entrevistas son esenciales en el proceso de la captura de requisitos y con su aplicación el equipo de desarrollo puede obtener una amplia visión del trabajo y las necesidades del usuario, es necesario destacar que no es una técnica sencilla de aplicar (15). Requiere que el entrevistador sea experimentado y tenga capacidad para elegir bien a los entrevistados y obtener de ellos toda la información posible en un período de tiempo siempre limitado. Aquí desempeña un papel fundamental la preparación de la entrevista (ver Técnica Cuestionarios y Listas de chequeo).
- **Desarrollo en conjunto de aplicaciones (JAD):** resulta una alternativa a las entrevistas. Es una práctica de grupo que se desarrolla durante varios días y en la que participan analistas, usuarios, administradores del sistema y clientes (16). Está basada en cuatro principios fundamentales: dinámica de grupo, el uso de ayudas visuales para mejorar la comunicación,

# Capítulo 1: Fundamentación Teórica

---

mantener un proceso organizado y racional y una filosofía de documentación WYSIWYG (What You See Is What You Get, lo que ve es lo que obtiene), es decir, durante la aplicación de la técnica se trabajará sobre lo que se generará. Tras una fase de preparación del JAD al caso concreto, el equipo de trabajo se reúne en varias sesiones. En cada una de ellas se establecen los requisitos de alto nivel a trabajar, el ámbito del problema y la documentación. Durante la sesión se discute en grupo sobre estos temas, llegándose a una serie de conclusiones que se documentan. En cada sesión se van concretando más las necesidades del sistema. Esta técnica presenta una serie de ventajas frente a las entrevistas tradicionales, ya que ahorra tiempo al evitar que las opiniones de los clientes se tengan que contrastar por separado, pero requiere un grupo de participantes bien integrados y organizados.

- **Tormenta de ideas:** es también una técnica de reuniones en grupo cuyo objetivo es que los participantes muestren sus ideas de forma libre (17). Consiste en la mera acumulación de ideas y/o información sin evaluar las mismas. El grupo de personas que participa en estas reuniones no debe ser muy numeroso (máximo 10 personas), una de ellas debe asumir el rol de moderador de la sesión, pero sin carácter de controlador. Como técnica de captura de requisitos es sencilla de usar y de aplicar, contrariamente al JAD, puesto que no requiere tanto trabajo en grupo como éste. Además suele ofrecer una visión general de las necesidades del sistema, pero normalmente no sirve para obtener detalles concretos del mismo, por lo que suele aplicarse en los primeros encuentros.
- **Mapas conceptuales:** son grafos en los que los vértices representan conceptos y las aristas representan posibles relaciones entre dichos conceptos (15). Estos grafos de relaciones se desarrollan con el usuario y sirven para aclarar los conceptos relacionados con el sistema a desarrollar. Son muy usados dentro de la Ingeniería de Requisitos, pues son fáciles de entender por el usuario, más aún si el equipo de desarrollo hace el esfuerzo de elaborarlo en el lenguaje de éste. Sin embargo, deben ser usados con cautela porque en algunos casos pueden ser muy subjetivos y pueden llegar a ser ambiguos en casos complejos, si no se acompaña de una descripción textual.
- **Casos de Uso:** aunque inicialmente se desarrollaron como técnica para la definición de requisitos, (12) algunos autores proponen casos de uso como técnica para la captura de requisitos (14). Los casos de uso permiten mostrar el contorno (actores) y el alcance (requisitos funcionales expresados como casos de uso) de un sistema. Un caso de uso describe la secuencia de interacciones que se producen entre el sistema y los actores del mismo para realizar una determinada función. Los actores son elementos externos (personas, otros

# Capítulo 1: Fundamentación Teórica

---

sistemas, entre otros) que interactúan con el sistema como si de una caja negra se tratase. Un actor puede participar en varios casos de uso y un caso de uso puede interactuar con varios actores. La ventaja esencial de los casos de uso es que resultan muy fáciles de entender para el usuario o cliente, sin embargo carecen de la precisión necesaria si no se acompañan con una información textual o detallada con otra técnica como pueden ser los diagramas de actividades. (18)

- **Cuestionarios y Listas de chequeo:** esta técnica requiere que el analista conozca el ámbito del problema en el que está trabajando. Consiste en redactar un documento con preguntas cuyas respuestas sean cortas y concretas, o incluso cerradas por unas cuantas opciones en el propio cuestionario (Listas de chequeo). Este cuestionario será respondido por el grupo de personas entrevistadas o simplemente para recoger información en forma independiente de una entrevista. (18)
- **Comparación de terminología:** uno de los problemas que surge durante la identificación de los requisitos es que usuarios y expertos no llegan a entenderse debido a problemas de terminología. Esta técnica es utilizada en forma complementaria a otras técnicas para obtener consenso respecto de la terminología a ser usada en el proyecto de desarrollo. (15)
- **Observación:** posibilita el entendimiento de los procesos del negocio a partir de observar la forma real de trabajar más que de la forma en la que se debería trabajar de acuerdo a los procesos establecidos. Permite identificar requisitos derivados de la cooperación y conocimiento de las actividades de las personas.
- **Arqueología de Documentos:** permite determinar posibles requisitos sobre la base de inspeccionar la documentación utilizada y generada en la empresa. Sirve fundamentalmente como complemento de las demás técnicas y mediante la misma se obtiene información que de otra forma sería imposible, como son manuales de procedimientos, reglamentos, facturas, entre otras. (17)

Existen más técnicas para la captura de requisitos (análisis de otros sistemas, entre otros), incluso también es común encontrar alternativas que combinen varias de estas técnicas (15). Sin embargo, las presentadas ofrecen un conjunto representativo de las más utilizadas y resultan suficientes para el objetivo de este trabajo.

## 1.7 Lenguajes y notaciones para la modelación de procesos.

En la actualidad existen un gran número de lenguajes, notaciones y herramientas de software que han sido diseñadas para la modelación y análisis de los procesos de negocio en este trabajo se muestran algunos de ellos. Un modelo puede dar la oportunidad de organizar y documentar la información sobre un sistema. Pero ¿qué es un modelo? Un modelo es una representación de una realidad compleja que permite desarrollar una descripción lo más exacta posible de un sistema y de las actividades llevadas a cabo en él. Cuando un proceso es modelado pueden apreciarse con facilidad las interrelaciones existentes entre distintas actividades, analizar cada actividad, definir los puntos de contacto con otros procesos, así como identificar los subprocessos comprendidos. Al mismo tiempo, los problemas existentes pueden ponerse de manifiesto claramente dando la oportunidad al inicio de acciones de mejora (19). A continuación se explican algunos lenguajes y notaciones para la modelación de procesos de negocio.

### **IDEF0: Definición de la integración para la modelización de las funciones.**

Lenguaje de modelado que constituye una técnica de modelación gráfica, especializada en la representación de las relaciones e interdependencias existentes entre los diferentes procesos, como se muestra en el esquema de la Figura 1.2. (20)

Su principal característica consiste en su capacidad para diferenciar entre tres tipos posibles de relación entre procesos: (20)

- Relaciones que establecen las guías que debe tener en cuenta el proceso.
- Relaciones que aportan los recursos necesarios para llevar a cabo el proceso.
- Relaciones de encadenamiento lineal entre procesos (entrada – salida).

La capacidad de diferenciar relaciones permite modelar organizaciones completas.



**Figura 1.2 Representación gráfica de un proceso en IDEF0.**

## 1.7.1 Cadena de proceso guiada por eventos

Es un modelo dinámico que representa juntos los recursos del negocio como son los sistemas, la organización, datos e información y los organiza para brindar una secuencia de tareas o actividades (el proceso) que añaden valor al negocio. (20)

Esencialmente hay cuatro tipos de objetos usados en este modelo:

- Eventos.
- Funciones.
- Reglas.
- Recursos (datos, organización, sistemas)

La filosofía básica en este tipo de modelos es representar una secuencia evento-función-evento-función-evento..., especificando para cada función las reglas y recursos que intervienen.

## 1.7.2 Lenguaje de Modelado Unificado (UML)

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables. Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema de software, para detallar los artefactos en el sistema, documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software, pero no especifica en sí mismo qué metodología o proceso usar. (21)

## 1.7.3 Notación para la Modelación de Procesos de Negocio (BPMN en sus siglas en inglés).

BPMN es una notación para el modelado de procesos de negocios a través de diagramas. Tiene como objetivo principal servir como soporte para la gestión por procesos, como una notación que pueda ser entendida fácilmente desde los analistas que crean los bocetos iniciales del proceso, los desarrolladores técnicos responsables de implementar la tecnología que ejecutará estos procesos, hasta las personas que los ejecutan y aquellas que llevaran a cabo el monitoreo y supervisión de los procesos. En otras palabras esta notación crea un enlace entre las etapas de diseño e implementación. (21) (22)

A pesar de ser intuitiva para todos los usuarios de negocio es capaz de representar semánticas de procesos complejos. Otro objetivo es asegurar que los lenguajes XML diseñados para la ejecución de procesos de negocio puedan ser visualizados con una notación común, dentro de los lenguajes se encuentra BPEL4WS, que es un lenguaje de ejecución de procesos de negocios para servicios web. (22)

## 1.8 Herramientas para el Modelado.

A través del modelado de procesos puede lograrse un mejor entendimiento de los negocios, creando la oportunidad de mejorarlos. Modelar la estructura organizativa de los procesos aporta visión global y permite comprender la dinámica del conjunto de relaciones de la organización, facilitando la alineación de actividad y la gestión de los cambios necesarios para disfrutar de flexibilidad. El modelado del negocio es la técnica por excelencia para alinear los desarrollos con las metas y objetivos de las empresas e instituciones y se realiza de tal forma en que el modelo quede consensuado entre los grupos interesados (7). El modelado de negocios, y más específicamente el modelado de procesos de negocio, es la forma idónea para comunicarnos con los usuarios de todos los niveles. El modelado de procesos constituye la base para el análisis, a partir del cual se identificarán los aspectos que tienen problemas y por tanto deben ser mejorados. A continuación se explican algunas herramientas para la modelación de procesos de negocio.

Las **herramientas CASE** (Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas nos pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño

dado, compilación automática, documentación o detección de errores entre otras (7). Entre sus principales objetivos se encuentran:

- Mejorar la productividad en el desarrollo y mantenimiento del software.
- Aumentar la calidad del software.
- Reducir el tiempo y coste de desarrollo y mantenimiento de los sistemas informáticos.
- Mejorar la planificación de un proyecto.
- Aumentar la biblioteca de conocimiento informático de una empresa ayudando a la búsqueda de soluciones para los requisitos.
- Automatizar el desarrollo del software, la documentación, la generación de código, las pruebas de errores y la gestión del proyecto.
- Ayuda a la reutilización del software, portabilidad y estandarización de la documentación.
- Gestión global en todas las fases de desarrollo de software con una misma herramienta.
- Facilitar el uso de las distintas metodologías propias de la ingeniería del software.

## 1.8.1 Rational Rose

Rational Rose es una herramienta de producción y comercialización establecidas por Rational Software Corporation (actualmente parte de IBM). Es un instrumento operativo conjunto que utiliza el Lenguaje Unificado (UML) como medio para facilitar la captura de dominio de la semántica, la arquitectura y el diseño. Este software tiene la capacidad de: crear, ver, modificar y manipular los componentes de un modelo. Otro aspecto a destacar es la posibilidad de generar y realizar ingeniería inversa en varios lenguajes de programación y el número de frameworks que tienen predefinidos, entre los cuales están J2EE, .Net, Visual Basic 6 y C++.

Entre sus características principales se encuentran: (23)

- No es gratuito, se debe hacer un previo pago para poder adquirir el producto.
- Incluye Java y la generación de clave de CORBA, la aplicación de ingeniería inversa y la sincronización de modelo para clave configurable.
- Admite la integración con otras herramientas de desarrollo (IDEs).
- Proporciona mecanismos para realizar la ingeniería inversa, además de que brinda la posibilidad del trabajo en equipo. (23) (24)
- Permite generar código en diferentes lenguajes a partir de un diseño en UML.
- Es un software propietario que soporta los diagramas de UML, excepto los diagramas de implementación.

- Propone la utilización de cuatro tipos de modelo para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico.

## 1.8.2 Visual Paradigm

Visual Paradigm es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una construcción más rápida de aplicaciones con calidad y con menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Es un potente conjunto de herramientas CASE UML que soporta UML y BPMN. También proporciona abundantes tutoriales de UML. (25)

Entre sus características principales se encuentran: (25)

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
- Modelado colaborativo con CVS y Subversion (nueva característica).
- Interoperabilidad con modelos UML2 (metamodelos UML 2.x para plataforma Eclipse) a través de XML (nueva característica).
- Ingeniería inversa - Código a modelo, código a diagrama, Ingeniería inversa Java, C++, Esquemas XML, XML, .NET, exe/dll, CORBA IDL.
- Generación de código - Modelo a código, diagrama a código.
- Editor de detalles de Casos de Uso, entorno "todo en uno" para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de uso.
- Diagramas de flujo de datos.
- Soporte de mapeo de objetos relacionales, generación de objetos Java desde la base de datos, generación de bases de datos, transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Ingeniería inversa de bases de datos desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación.
- Generador de informes para generación de documentación.
- Distribución automática de diagramas, reorganización de las figuras y conectores de los diagramas UML.
- Importación y exportación de ficheros XML.

- Integración con Visio, dibujo de diagramas UML con plantillas de MS Visio.

### 1.8.3 Enterprise Architect

Proporciona un entorno de modelización de carácter colaborativo y potenciado mediante UML 2.1, abarca por completo el ciclo de vida de desarrollo de software, con herramientas que le proporcionan una infraestructura enormemente competitiva en torno a la modelización de negocio, diseño de software, ingeniería de sistemas, arquitectura corporativa, gestión de requisitos, pruebas. Presenta las siguientes características: (23)

- Comprensivo para UML 2.1 y soporta los 13 diagramas de UML 2.1.
- Interfaz de usuario intuitiva. Amplio rango de barras de herramientas, ventanas acoplables, y estilos visuales.
- Documentación flexible y comprensible.
- Ingeniería de código directa e Inversa. Generador de código dirigido por plantillas.
- Tecnologías agregadas usando tecnologías MDG (Generación dirigida por modelos).
- Modelado de base de datos.
- Ingeniería inversa para muchos de los sistemas populares DBMS, incluyendo Oracle, SQL Server, My SQL, Access, Postgre SQL y otros.
- Capacidad para compartir modelos de diversas maneras para control de versiones.
- Soporta esquemas XML.
- Utilidad de Comparación.
- Archivos binarios de ingeniería inversa para Java y .NET.
- Administración de requisitos.
- Importación / Exportación de modelos en formato XML.
- Seguridad de Usuario.
- Permite realizar Pruebas de Unidad, Pruebas de Integración, Pruebas de sistema, Pruebas de aceptación.

### 1.8.4 Justificación de la Herramienta de Modelado Seleccionada.

Teniendo en cuenta las características antes expuestas de las herramientas analizadas y realizando un profundo estudio de las mismas, se llega a la conclusión de que se utilizará Visual Paradigm como herramienta de desarrollo de software ya que es multiplataforma, posee alta capacidad de integración

# Capítulo 1: Fundamentación Teórica

---

con el lenguaje orientado a procesos BPMN y UML que serán los utilizados para el desarrollo del sistema. La UCI posee una licencia que utiliza sin fines de lucro y propone el uso de esta robusta herramienta para el desarrollo de los proyectos productivos. Además el equipo de desarrollo cuenta con un mayor conocimiento de esta herramienta que de otras descritas, posibilitando ahorro de tiempo en el desarrollo del software ofreciendo capacitación de los desarrolladores.

## **1.9 Herramientas para Modelado de Prototipos no Funcionales.**

En el presente epígrafe se realizará un estudio de las herramientas que se usan en la actualidad para la confección de los prototipos no funcionales.

### **1.9.1 Microsoft Office Visio 2007.**

Office Visio 2007 permite documentar, diseñar y comprender de forma visual el estado de los sistemas y procesos empresariales con una gran variedad de diagramas, diagramas de flujo de proceso empresarial, diagramas de red, diagramas de flujo de trabajo, modelos de bases de datos y diagramas de software, entre otros. (26)

Permite conectar automáticamente diagramas con uno o varios orígenes de datos como hojas de cálculo de Microsoft Office Excel 2007 o bases de datos de Microsoft Office Access 2007.

Dentro de las ventajas que proporciona esta herramienta para el modelado de requisitos no funcionales vale destacar: (26)

- Permite documentar y diseñar los procesos y los sistemas para ofrecer una información completa y actualizada.
- Explora la información compleja para identificar tendencias, excepciones y puntos de vista clave.
- Comparte visualmente la información compleja para poder tomar decisiones más eficaces para la empresa.

### **1.9.2 BPWin 4.1**

BPwin es una potente herramienta para analizar, documentar y mejorar los procesos de negocio de la organización, permite documentar de manera clara las actividades que son necesarias, cómo se realizan y qué recursos consumen. Proporciona un marco de trabajo para poder representar y entender los procesos de negocio, determinando el impacto de los diferentes sucesos y definiendo cómo los

# Capítulo 1: Fundamentación Teórica

---

procesos interactúan unos con otros mediante flujos de información posibilitando identificar actividades poco eficientes o redundantes (27). Entre sus beneficios se encuentran:

- **Diseño automatizado de procesos:** permite corregir o verificar la consistencia de los resultados en el diseño del proceso.
- **Propiedades definidas por el usuario:** se puede personalizar para capturar información relevante a sus procesos y utilizarlas cuando lo requiera.
- **Técnicas de integración:** provee las técnicas para documentar los procesos empresariales y flujos de proceso a través del IDEF0, IDEF3 y DFD.
- **Métrica y análisis de costos:** actividades basadas en costos para la maximización de beneficios en los procesos pueden ser creadas de manera simple y sencilla.
- **Pre evaluación:** ofrece una interfaz para la simulación del software con la finalidad de explorar los efectos producidos en el negocio debido a los cambios realizados en el sistema.
- **Explorador de modelos:** incluye un explorador que permite visualizar de manera alterna los objetos creados o la ramificación de un diagrama elegido con la finalidad de acceder más rápido a cada uno de ellos.
- **Diccionarios:** todos los diccionarios poseen una interfaz gráfica formada por celdas y columnas parecida a una hoja de cálculo. Dichos diccionarios pueden ser personalizados con la finalidad de especificar sus necesidades de impresión, exportación o importación.
- **Diálogo de propiedades:** tanto los diagramas como cada uno de los elementos que participan en él, poseen una caja de diálogo con la finalidad de establecer propiedades relacionadas a color, tamaño, tipo de gráfico, cabeceras y pies del diagrama, configuración de página.
- **Asociación de entidades y datos:** se puede asociar las entidades creadas en BPwin con la herramienta Erwin al momento de exportarlas.

### 1.9.3 Axure RP Pro

Axure RP Pro es la principal herramienta para la rápida creación de prototipos y especificaciones para aplicaciones y sitios web. Demuestra su grado de especialización en las anotaciones, en este punto, permite especificar el estado de cada elemento (Propuesto, Aceptado, Incorporado), el beneficio esperado (Crítico, Importante, Útil), el riesgo, la estabilidad, a quién va dirigido y a quién se le asignará la tarea. (28)

## Características Principales de Axure RP Pro:

- Fácil de usar.
- Permite realizar cambios muy rápidos en la aplicación a diseñar.
- Crea rápidamente prototipos funcionales interactivos y con anotaciones.
- Versión de administración.
- Creación de un diseño efectivo.
- Permite experimentar con su propio diseño para lograr los máximos resultados.
- Automatización de la especificación para evitar la tediosa documentación.

### 1.9.4 Justificación de la Herramienta para el Modelado de Prototipos no Funcionales seleccionada.

Analizando las características antes expuestas para la confección de prototipos no funcionales se decide utilizar la herramienta Axure RP Pro en versión 5.0, ya que muestra los prototipos de una forma dinámica, permite al usuario poder interactuar con formularios y de esta manera tener una idea más clara de cómo será la aplicación web en un futuro.

### 1.10 Patrones de Casos de Uso

Los patrones de software describen un problema que ocurre repetidas veces en algún contexto determinado del proceso de desarrollo de software, y entregan una buena solución ya probada.

Esto ayuda a diseñar correctamente en menos tiempo y a construir problemas reutilizables y extensibles, y facilita la documentación y la comunicación con otros miembros del equipo de desarrollo (29). Una vez modelados los procesos de negocio, estos deben facilitar la identificación de los requisitos del sistema. Para ello, a continuación se proponen una serie de patrones y guías que facilitan su identificación como Casos de Uso (CU) (30). La secuencia de actividades a realizar para ello serían las siguientes:

- Identificar los actores del sistema. Un actor será todo agente externo al negocio, los agentes internos del negocio o sistemas heredados.
- Marcar las actividades de los procesos que deben estar soportadas por el sistema, marcando cada actividad automática, soportada o manual.
- Identificar los casos de uso a partir de los procesos de negocio. Para lograrlo, se deben detectar los conjuntos de tareas que pueden ser realizadas por un actor sin interrupciones.

# Capítulo 1: Fundamentación Teórica

---

**Concordancia – Adición:** En el caso de este patrón alternativo, la subsecuencia común de casos de uso, extiende los casos de uso compartiendo la subsecuencia de acciones. Los otros casos de uso modelan el flujo que será expandido con la subsecuencia. Este patrón es preferible usarlo cuando otros casos de uso se encuentran propiamente completos, o sea, que no requieren de una subsecuencia común de acciones para modelar los usos completos del sistema. (29) (30)

**Concordancia – Reuso:** Consta de 3 casos de uso. El primero llamado subsecuencia común, modela una secuencia de acciones que aparecerán en múltiples casos de uso en el modelo. Los otros casos de uso modelan el uso del sistema que comparte la subsecuencia común de acciones. De manera que deben existir al menos dos de ellos. A continuación se muestra un ejemplo de su uso. (29) (30)

**Inclusión Concreta:** es un patrón de estructura. Consiste en dos casos de uso y una relación de inclusión entre el caso de uso base y el caso de uso incluido. Este último puede ser instanciado por sí solo. El caso de uso base puede ser concreto o abstracto. Se utiliza este patrón cuando un flujo de datos puede ser incluido en el flujo de otro caso de uso y también puede ejecutarse por sí solo. (29) (31)

**Extensión Concreta:** este patrón consiste en dos casos de uso y una relación de extensión entre ellos. El caso de uso extendido es concreto, es decir, este puede ser instanciado por sí solo, así como, ser una extensión del caso de uso base. El caso de uso base puede ser concreto o abstracto. Este patrón es aplicable cuando un flujo de datos puede ser extendido del flujo de datos de otro caso de uso, así como ser ejecutado por sí solo. (29) (31)

**CRUD- Completo (Crear, Modificar, Eliminar, Mostrar):** este patrón se basa en la fusión de casos de uso simples para formar una unidad conceptual. Consta de un caso de uso, llamado Información CRUD o Gestionar información que modela todas las operaciones que pueden ser realizadas sobre una parte de la información de un tipo específico, tales como creación, lectura, actualización y eliminación (31). Suele ser utilizado cuando todos los flujos contribuyen al mismo valor del negocio, y estos a su vez son cortos y simples. Existe un patrón alternativo llamado CRUD Parcial, que modela una de las vías de los casos de uso como un caso de uso separado. Es preferiblemente utilizado cuando una de las alternativas de los casos de uso es más significativa, larga o más compleja que las otras. (12)

**Actores múltiples: Rol común:** es un patrón de estructura que plantea que cuando dos actores juegan el mismo papel hacia un caso de uso se representa otro actor, del que heredan los actores que comparten este rol. Este patrón es aplicable cuando, desde el punto de vista de un caso de uso, hay solo una entidad externa interactuando con cada instancia del caso de uso. (31)

## 1.11 Métricas para Requisitos y Casos de Uso

Las métricas son un buen medio para entender, monitorizar, controlar, predecir y probar el desarrollo software y los proyectos de mantenimiento. En general, la medición persigue tres objetivos fundamentales: ayuda a entender qué ocurre durante el desarrollo y el mantenimiento, permite controlar qué es lo que ocurre en un proyecto y nos provee de poder mejorar nuestros procesos y nuestros productos. (7)

Para la medición de la calidad de la especificación de requisitos de un sistema de software se tienen en cuenta una lista de características que pueden emplearse para valorar la calidad del modelo de análisis y la correspondiente especificación de requisitos: Especificidad y dentro de ella la ausencia de ambigüedad, corrección, completión, comprensión, capacidad de verificación, consistencia externa e interna, capacidad de logro, concisión, trazabilidad, capacidad de modificación, exactitud y capacidad de reutilización (7). Además las especificaciones están empaquetadas por importancia y estabilidad relativas, con su versión correspondiente, organizadas, con referencias cruzadas y especificadas al nivel correcto de detalle. Aunque muchas de las características anteriores pueden ser de naturaleza cuantitativa, se sugiere que todas puedan representarse usando una o más métricas. (32)

$$NR = RF + RNF$$

Donde NR es el número de requisitos funcionales y RNF es el número de requisitos no funcionales.

Para determinar la especificidad de los requisitos, sugiere una métrica basada en la consistencia de la interpretación de los revisores para cada requisito:

$$Q1 = NRI / NR$$

Donde NRI es el número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas. Cuanto más cerca de 1 este el valor de Q1 menor será la ambigüedad de la especificación.

### **Métricas para la Calidad de los Casos de Uso.**

Validar el Diagrama de Caso de Uso del Sistema (DCUS) contribuye a estar seguro que los procesos descritos en el modelo se relacionan de la forma adecuada, permite refinar los casos de usos y sus relaciones, además de proporcionarle una categoría de acuerdo a su complejidad y a la importancia en

# Capítulo 1: Fundamentación Teórica

---

que se ejecutan para la aplicación. Una de las métricas más usadas a la hora de medir un DCUS es la de calidad de la funcionalidad del diagrama de casos de uso de sistema, donde se definen cuatro atributos. Dichos atributos son analizados por un conjunto de factores en varias iteraciones, cada uno de los cuales tiene asociada una o varias métricas, donde estas tomarán un valor en por ciento. Los atributos a medir son: (32)

- **Compleitud:** permite determinar el grado en que se ha incluido de forma clara y concisa todos los elementos necesarios para la descripción.
- **Consistencia:** permite definir el grado en que los elementos del DCUS representan en forma única y no contradictoria un aspecto del problema.
- **Correctitud:** permite establecer el grado de adecuación del DCUS para satisfacer los requisitos.
- **Complejidad:** permite medir el grado de claridad y re-uso del DCUS.

## 1.12 Conclusiones parciales

El análisis de algunas soluciones informáticas existentes que gestionan los procesos judiciales, demuestran la necesidad de crear el Módulo Económico para el Sistema de Informatización de Tribunales, debido a que estas no se ajustan a las leyes que rigen al procedimiento Económico, ni a las necesidades existentes en el Tribunales Populares Cubanos.

El estudio de las diferentes metodologías de desarrollo de software, lenguaje de modelado, herramientas CASE y herramientas para el modelado de prototipos no funcionales, permitió conocer cuáles son las que más se ajustan para el desarrollo de dicho módulo. En este caso las seleccionadas fueron: RUP, UML y BPMN, Visual Paradigm y Axure RP Pro. También se realizó un estudio de las técnicas para la captura de los requisitos del procedimiento Diligencias Previas, los patrones de casos de usos para el modelado del diagrama de casos de usos y las métricas para validar los requisitos y el Diagrama de casos de uso del sistema, sentándose las bases para comenzar la modelación del procedimiento Diligencias Previas del Módulo Económico.

## CAPÍTULO 2: SOLUCIÓN PROPUESTA

### 2.1 Introducción

A partir del estudio realizado en el Capítulo 1 de este trabajo y luego de haber identificado los métodos y herramientas que se utilizarán para la realización del análisis y modelado del negocio, se está en condiciones para ejecutar las acciones que permitan realizar del análisis del procedimiento Diligencias Previas del Módulo Económico del proyecto Tribunales Populares Cubanos.

Entre los propósitos fundamentales de este capítulo se encuentran: describir y brindar una mejor comprensión de dicho procedimiento, identificar quiénes participan y las actividades que requieran de informatización, se obtendrá como resultado los artefactos: flujograma de procesos, reglas del negocio, requisitos funcionales, requisitos no funcionales y el modelo de casos de uso del sistema.

### 2.2 Descripción del Proceso de Negocio de Diligencias Previas de la Materia Económica

Los Tribunales Populares Cubanos están divididos en 3 instancias: Supremo, Provincial y Municipal, donde son atendidas 5 materias: Laboral, Penal, Civil, Administrativo y Económico. La Materia Económica está compuesta por tres procedimientos: demandas por incumplimientos contractuales, diligencias previas y los conflictos de carácter económico que se susciten sobre la protección del medio ambiente y del uso racional de los recursos naturales.

A partir de la promulgación del Decreto Ley 223/2001, dada la experiencia acumulada por las Salas de lo Económico de los Tribunales Provinciales Populares y con la extinción del Sistema de Arbitraje Estatal, se transfieren de forma general a estas Salas los asuntos que habían sido conocidos por la Sala de lo Económico del Tribunal Supremo Popular. Se reservó a esta última únicamente los procesos extraordinarios de revisión y la facultad de resolver los conflictos de competencia por razón del territorio que puedan surgir entre las Salas de lo Económico de los Tribunales Provinciales Populares de los distintos territorios.

El procedimiento Diligencias Previas se inicia cuando el Abogado o la Secretaria de la Sala de lo Económico de los Tribunales Populares Cubanos registran un escrito promocional, este es admitido por un juez ponente de la sala notificándole a la Secretaria su admisión. Esta última, procede a citar a las partes involucradas en el litigio y cuando estas se presentan, la secretaria procede a realizar el acta de comparecencia entre ellas. Luego el Juez Ponente comienza a preparar la acción ejecutiva, y una vez ejecutada esta, la secretaria procede a archivar el expediente generado con la presentación del

## Capítulo 2: Solución Propuesta

escrito inicial. En la Figura 2.1 se representan brevemente algunos de los subprocesos más importantes.



**Figura 2.1 Breve descripción del procedimiento Diligencias Previas.**

Se refleja una mejor descripción en la siguiente Ficha de procesos, donde se explica de manera clara los subprocesos principales:

Ficha de Proceso	
Proceso:	Diligencias Previas al Proceso Ejecutivo
Responsable:	Juez Ponente
Ejecutor (es):	Secretaria Juez Ponente Presidente de la Sala
Descripción:	El proceso de diligencias previas tiene por objetivo preparar un título ejecutivo que le permita al acreedor hacer efectivo el pago de una deuda reconocida, con la mayor inmediatez. Concluida la diligencia previa queda preparada la acción ejecutiva, que puede dar paso a un proceso ejecutivo. No siempre sucede aún cuando es esta su razón de ser.
Entradas:	Escritos de Presentación: <ul style="list-style-type: none"> <li>• Escritos Promocionales</li> <li>• Escritos de Personería</li> <li>• Escritos Conciliatorios</li> </ul>
Salidas:	Autos

## Capítulo 2: Solución Propuesta

Procesos asociados:	Proceso Ejecutivo
Reglas del Negocio:	<p>1- El presunto acreedor tiene a lo sumo 5 días para subsanar el error.</p> <p>2- Una vez que se archive el expediente no se puede modificar.</p> <p>3- El expediente va foliado o paginado en orden consecutivo en la medida en que se generan los documentos.</p> <p>4- El presidente de la sala es quien designa al juez ponente.</p> <p>5- Las partes tienen 2 días como máximo para notificarse en mesa.</p> <p>6- Las notificaciones en mesa se firman.</p> <p>7- Las notificaciones en tablilla se realizan después de que se cumpla el término de las notificaciones en mesa, para ello hay 5 días.</p> <p>8- Cada vez que sale un documento del tribunal se asienta en el Cuaderno de salida de documentos (CSD).</p>
Clasificación:	Principal
Flujo básico:	<p>1-2-3-4-5-6-7-8-9-10-11-12-13-14-15</p> <p>1-2-3-4-5-6-7-8-16-14-15</p> <p>1-2-3-4-5-6-7-8--9-10-11-12-23</p>
<b>Actividades Principales</b>	
1. Presentar Escrito:	<p>Esta actividad empieza cuando el presunto acreedor presenta en la sala el escrito, que puede ser un escrito promocional, escrito de personería o escrito conciliatorio. No es más que el acto de presentarse frente a la secretaria con un documento.</p> <p>En el expediente los escritos van en el siguiente orden:</p> <ol style="list-style-type: none"> <li>1- Escrito promocional.</li> <li>2- Escrito de personería(a este se le aplica un cuño).</li> <li>3- Escrito conciliatorio (opcional).</li> </ol> <p>El expediente va foliado o paginado consecutivamente en la medida en que se generan los documentos.</p> <p>El escrito va acompañado de tres copias:</p> <ul style="list-style-type: none"> <li>• 1 para el abogado</li> <li>• 1 para el tribunal</li> <li>• 1 para el deudor</li> </ul> <p>Las copias son para dárselas al presunto deudor, son entregadas en el</p>

## Capítulo 2: Solución Propuesta

	<p>momento de la confesión. Mientras se guardan en el expediente, en la parte de atrás.</p>
2. Recibir Escrito:	<p>La secretaria recibe el documento traído por el presunto acreedor que bien puede ser un escrito promocional, un escrito de personería o un escrito conciliatorio. De ahí pasa a asentarlos en el Libro de presentación de escritos (LPE). Dicho escrito se adjunta al expediente descosándolo y volviéndolo a coser. El escrito va acompañado de tres copias:</p> <ul style="list-style-type: none"><li>• 1 para el abogado</li><li>• 1 para el tribunal</li><li>• 1 para el deudor</li></ul> <p>Las copias son para dárselas al presunto deudor, son entregadas en el momento de la confesión. Mientras se guardan presillados en la parte de atrás del expediente.</p>
3. Asentar Escrito en el LPE:	<p>Es cuando la secretaria asienta en el LPE la constancia oficial de la presentación de escritos y documentos acompañados. Este libro ya está creado, y cada vez que se presenta un nuevo escrito este se actualiza. Este libro es foliado consecutivamente. Aquí se registran los documentos que entregan las partes.</p>
4. Crear Expediente:	<p>La secretaria llena la carátula del expediente que se va a crear en cuestión. Seguidamente cose los escritos presentados a la dicha carátula quedando formado el expediente.</p>
5. Designar Juez Ponente:	<p>El presidente de la sala es el encargado de designar al juez ponente para la tramitación judicial. Se refiere a las políticas de turnado que se van a seguir en la sala. Se crea la providencia de turnado.</p>
6. Ejecutar Designación:	<p>El presidente de la sala es el encargado de designar al juez ponente pero la secretaria es la que ejecuta la designación creando la providencia de turnado para entonces asentar en el Libro de registro de escritos (LRE).</p>
7. Asentar en el LRE:	<p>La secretaria asienta en el LRE por su número consecutivo el expediente para su tramitación. Cada vez que se asienta en el LRE se actualiza el expediente y el LRE. Aquí es donde se le da el número al expediente.</p>

## Capítulo 2: Solución Propuesta

8. Notificar Expediente:	La secretaria le lleva el expediente al presidente de la sala para que este revise la providencia que ella ejecutó.
9. Admitir Escrito:	El escrito se admite cuando es de su jurisdicción, cuando es de su competencia y cuando no tiene causas de inadmisión. Si por casualidad no es de su jurisdicción o si no es de su competencia entonces se dictará rechazo de plano y le entregará el auto a la secretaria. Si tuviera causas de inadmisión entonces se dictará el auto de inadmisión y se le entregaría el auto a la secretaria, al igual, si no tiene causas de inadmisión, entonces se verifica si los errores son formales o no, si lo son entonces se dispone subsanación, si no tiene errores entonces se admite el escrito y se entrega la resolución. Puede admitir el escrito mediante providencia o auto sin número.
10. Notificar Resolución:	El juez ponente le notifica a la secretaria la resolución y esta entonces crea la diligencia de citación para citar a las partes.
11. Enviar Cédula de Citación:	La secretaria crea la cédula de citación y se la envía al presunto deudor para que este comparezca y reconozca la firma y confiese la deuda o no. Esta cédula se archiva en el Libro de salidas de documentos (LSD). Se crea una resulta que no es más que la respuesta del presunto deudor a dicha citación, también se le aplica un cuño y se crea un modelo de notificación. Esto se envía por la correspondencia del tribunal, lo hace el alguacil. En la práctica se cita también al presunto acreedor, generalmente es el que comparece. Su presencia o ausencia no repercute para nada en el proceso. Si la citación fue enviada y después la parte promovente va al tribunal (mientras la citación le está llegando), la secretaria le hace otra citación.
12. Redactar Acta de Comparecencia	<p>La secretaria redacta el acta en el momento del acto.</p> <p>Pueden ser 4 tipos de acta según el caso:</p> <ol style="list-style-type: none"> <li>1- Comparecieron las dos partes</li> <li>2- No comparecieron ninguna de las dos partes</li> <li>3- Solo compareció la parte deudora</li> <li>4- Solo compareció la parte acreedora</li> </ol>

## Capítulo 2: Solución Propuesta

13. Preparar la Acción Ejecutiva:	Este auto es creado cuando el presunto deudor no se presenta a comparecer o cuando se presenta y reconoce la firma o confiesa la deuda y entonces ocurre el subproceso concusión y archivo (ver Subproceso Conclusión y Archivo).
14. Notificar Auto:	El juez ponente entrega el auto a la secretaria y ocurre el subproceso conclusión y archivo (ver Subproceso Conclusión y Archivo). Puede ser la providencia o el auto, en dependencia de la resolución con la cual admitió el escrito.
15. Conclusión y Archivo:	Subproceso Conclusión y Archivo que se verá más adelante.
16. Dictar Rechazo de Plano:	El juez ponente dicta el rechazo de plano cuando la tramitación no es de su jurisdicción o de su competencia y se entrega el auto a la secretaria y ocurre el subproceso conclusión y archivo (ver Subproceso Conclusión y Archivo).
17. Dictar ANHLA:	Es cuando es de su jurisdicción y de su competencia se verifica si tiene causas de inadmisión o no, si las tiene entonces se dictará el auto no haber lugar admitir y se notifica el auto a la secretaria y ocurre el subproceso Conclusión y Archivo.
18. Disponer Subsanación:	Esto sucede cuando el escrito tiene errores formales, el juez ponente dispone la subsanación de estos y la secretaria notifica dichos errores al acreedor. Una vez subsanados éstos, se asienta el escrito en el LPE, sino se han subsanado, entonces se dictará auto de inadmisión.
19. Notificar Subsanación:	<p>La secretaria notifica dichos errores al acreedor. Una vez subsanados estos, se asienta el escrito en el LPE, sino se han subsanado, entonces se dictará auto de inadmisión.</p> <p>Las notificaciones se harán de la siguiente forma y orden de prioridades:</p> <ol style="list-style-type: none"> <li>1- En mesa (personalmente): dos días máximos. El abogado firma.</li> <li>2- En tablilla: después que pasen los días de mesa, tienen 5 días para tablilla. No se firma.</li> <li>3- A domicilio.</li> </ol> <p>En este caso la subsanación la lee el abogado en el expediente, porque está en la obligación de ir todos los días al tribunal. Si no lo hace, se procede a los otros medios de notificación.</p>

## Capítulo 2: Solución Propuesta

	<p>Cuando se notifica la subsanación se le da copia de la providencia a las partes.</p>
20. Subsananar el Error:	<p>Es cuando el presunto acreedor recibe de parte de la secretaria el escrito que tiene los errores formales para que este en un término de 5 días a lo sumo, a partir del día siguiente en que fue notificado, subsane dichos errores. Si los subsana entonces la secretaria asienta el escrito en el LPE, sino se han subsanado, entonces se creará un auto de inadmisión.</p>
21. Asentar Escrito:	<p>Si los errores son subsanados la secretaria asienta el escrito en el LPE y lo actualiza. Hace un dar cuenta y se lo envía al juez ponente, el cual creará un auto de inadmisión y entrega el auto. Este modelo también es archivado en el expediente.</p>
22. Dar Cuenta:	<p>La secretaria hace este dar cuenta cuando los errores son subsanados para entregárselo al juez ponente y éste creará un auto de inadmisión y entregará el auto. El expediente es descosido y se adjunta dicho modelo y se vuelve a coser. Si cuando llega el documento subsanado el juez ponente no está (por las razones que fueran), la secretaria da cuenta al presidente de sala y éste mediante providencia retorna el caso a otro juez.</p>
23. Dictar Auto Denegando a la Acción Ejecutiva:	<p>Este auto se crea cuando el presunto deudor se presenta a comparecer y no reconoce la firma ni confiesa la deuda.</p>
24. Preparar la Acción Ejecutiva:	<p>Este auto es creado cuando el presunto deudor no se presenta a comparecer o cuando se presenta y reconoce la firma o confiesa la deuda y entonces ocurre el subproceso concusión y archivo (ver Subproceso Conclusión y Archivo).</p>
25. Notificar Auto:	<p>El juez ponente entrega el auto a la secretaria y ocurre el subproceso conclusión y archivo (ver Subproceso Conclusión y Archivo). Puede ser la providencia o el auto, en dependencia de la resolución con la cual admitió el escrito.</p>
26. Conclusión y Archivo:	<p>Subproceso Conclusión y Archivo que se verá más adelante.</p>

## Capítulo 2: Solución Propuesta

27. Dictar Rechazo de Plano:	El juez ponente dicta el rechazo de plano cuando la tramitación no es de su jurisdicción o de su competencia y se entrega el auto a la secretaria y ocurre el subproceso conclusión y archivo (ver Subproceso Conclusión y Archivo).
28. Dictar ANHLA:	Es cuando es de su jurisdicción y de su competencia se verifica si tiene causas de inadmisión o no, si las tiene entonces se dictará el auto no haber lugar admitir y se notifica el auto a la secretaria y ocurre el subproceso Conclusión y Archivo.
29. Disponer Subsanación:	Esto sucede cuando el escrito tiene errores formales, el juez ponente dispone la subsanación de estos y la secretaria notifica dichos errores al acreedor. Una vez subsanados éstos, se asienta el escrito en el LPE, sino se han subsanado, entonces se dictará auto de inadmisión.
30. Notificar Subsanación:	<p>La secretaria notifica dichos errores al acreedor. Una vez subsanados estos, se asienta el escrito en el LPE, sino se han subsanado, entonces se dictará auto de inadmisión.</p> <p>Las notificaciones se harán de la siguiente forma y orden de prioridades:</p> <ol style="list-style-type: none"> <li>1- En mesa (personalmente): dos días máximos. El abogado firma.</li> <li>2- En tablilla: después que pasen los días de mesa, tienen 5 días para tablilla. No se firma.</li> <li>3- A domicilio.</li> </ol> <p>En este caso la subsanación la lee el abogado en el expediente, porque está en la obligación de ir todos los días al tribunal. Si no lo hace, se procede a los otros medios de notificación.</p> <p>Cuando se notifica la subsanación se le da copia de la providencia a las partes.</p>
31. Subsanan el Error:	Es cuando el presunto acreedor recibe de parte de la secretaria el escrito que tiene los errores formales para que este en un término de 5 días a lo sumo, a partir del día siguiente en que fue notificado, subsane dichos errores. Si los subsana entonces la secretaria asienta el escrito en el LPE, sino se han subsanado, entonces se creará un auto de inadmisión.

## Capítulo 2: Solución Propuesta

32. Asentar Escrito:	Si los errores son subsanados la secretaria asienta el escrito en el LPE y lo actualiza. Hace un dar cuenta y se lo envía al juez ponente, el cual creará un auto de inadmisión y entrega el auto. Este modelo también es archivado en el expediente.
33. Dar Cuenta:	La secretaria hace este dar cuenta cuando los errores son subsanados para entregárselo al juez ponente y éste creará un auto de inadmisión y entregará el auto. El expediente es descosido y se adjunta dicho modelo y se vuelve a coser. Si cuando llega el documento subsanado el juez ponente no está (por las razones que fueran), la secretaria da cuenta al presidente de sala y éste mediante providencia retorna el caso a otro juez.
34. Dictar Auto Denegando a la Acción Ejecutiva:	Este auto se crea cuando el presunto deudor se presenta a comparecer y no reconoce la firma ni confiesa la deuda.

Para obtener información detallada del flujograma de procesos se puede consultar el documento: **Modelo del Negocio con BPMN**, adjunto a la investigación.

### 2.3 Requisitos del Sistema

Los requisitos de un software se especifican por escritos de la misma manera en que se realiza para firmar un contrato o un acuerdo entre dos partes o más. Tiene que existir una forma para verificarlo o comprobarlo y así solo se obtendrá la veracidad del mismo. Un requisito de cualquier índole es la manera de justificar lo que el sistema debe hacer y no como debe hacerlo, y deben estar descritos como una característica del sistema a entregar. Es muy importante destacar que deben estar redactados de la manera más clara y concisa posible para evitar malas interpretaciones entre las partes involucradas en el sistema. Existen dos formas de agrupar los requisitos de una aplicación: Requisitos Funcionales (RF) y los Requisitos no Funcionales (RNF); los primeros son capacidades o condiciones que el sistema debe cumplir: “futuras opciones, acciones ocultas y condiciones extremas a determinar por el software” y los segundos son propiedades o cualidades que el producto debe tener: “atractivo, usable, rápido y confiable”.

Para la obtención de los requisitos funcionales del procedimiento Diligencias Previas del Módulo Económico se utilizaron algunas de las técnicas de captura de requisitos estudiadas en el Capítulo 1. En las primeras reuniones con los clientes se llevó a cabo la técnica “Tormenta de Ideas” para obtener una panorámica general del procedimiento Diligencias Previas y además se identificaron las

necesidades fundamentales que el sistema debía cumplir, estas primeras ideas se fueron concretando mediante las “Entrevistas, en las cuales se detallaron cada una de las funcionalidades que el sistema debía cumplir. En una etapa más avanzada se aplicó la técnica de “Desarrollo en conjunto de Aplicaciones”, haciendo un análisis exhaustivo de los documentos y obteniéndose los requisitos.

### 2.3.1 Requisitos Funcionales

Para consultar información detallada de los requisitos funcionales obtenidos ver documento: **Especificación de Requisitos Funcionales de Software**, adjunto a la investigación.

**RF.01** Registrar escrito.

**RF.02** Registrar escrito promocional por secretaria.

**RF.03** Registrar escrito promocional por abogado.

**RF.04** Asignar número de expediente.

**RF.05** Asignar expediente.

**RF.06** Crear providencia de turnado.

**RF.07** Notificar resolución.

**RF.08** Registrar acción.

**RF.09** Reasignar expediente.

**RF.10** Crear providencia de returnado.

**RF.11** Modificar resolución.

**RF.12** Exportar resolución a PDF.

**RF.13** Buscar expediente.

**RF.14** Modificar composición del tribunal.

**RF.15** Crear providencia de Admisión y Señalamiento.

**RF.16** Crear diligencia de citación.

**RF.17** Registrar notificación.

**RF.18** Registrar resulta.

**RF.19** Crear acta de comparecencia.

**RF.20** Crear Auto preparando acción ejecutiva.

- RF.21** Crear Auto no preparando acción ejecutiva.
- RF.22** Crear providencia de subsanación.
- RF.23** Crear Auto de rechazo de plano.
- RF.24** Crear Auto no haber lugar a admitir.
- RF.25** Crear auto declarando extemporáneo.
- RF.26** Crear providencia de nuevo señalamiento.
- RF.27** Visualizar escritos presentados.
- RF.28** Visualizar expedientes radicados.
- RF.29** Visualizar registro de autos numerados.
- RF.30** Visualizar cantidad de expedientes por monto.
- RF.31** Visualizar cantidad de expedientes resueltos.
- RF.32** Visualizar cantidad de expedientes fuera de término.
- RF.33** Visualizar cantidad de expedientes en trámite.
- RF.34** Visualizar cantidad de expedientes radicados por confesión de deuda.
- RF.35** Visualizar cantidad de expedientes radicados por reconocimiento de firma.
- RF.36** Visualizar cantidad de expedientes inadmitidos.
- RF.37** Visualizar cantidad de expedientes desistidos.
- RF.38** Visualizar cantidad de expedientes preparados.
- RF.39** Visualizar cantidad de expedientes denegados.
- RF.40** Visualizar cantidad de expedientes pendientes al inicio.
- RF.41** Visualizar cantidad de expedientes pendientes al final.
- RF.42** Visualizar acciones.
- RF.43** Mostrar cantidad de expedientes en término.
- RF.44** Informar de escrito presentado.
- RF.45** Visualizar documentos.
- RF.46** Crear auto resolviendo recurso de súplica.
- RF.47** Buscar resolución.

**RF.48** Crear providencia de archivo.

**RF.49** Crear auto de desistimiento.

**RF.50** Visualizar calendario de señalamientos.

**RF.51** Modificar escrito.

### 2.3.2 Requisitos no Funcionales

Los requisitos No Funcionales fueron agrupados según la categoría en la que fueron definidos.

- **Requisitos de Usabilidad**

**RNF.01** 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.

**RNF.02** Existirán servidores locales con capacidad necesaria para el procesamiento de las solicitudes del conjunto de aplicaciones de las diferentes oficinas.

**RNF.03** La aplicación siempre solicitará los datos a través del servidor local.

**RNF.04** Desde cada servidor local se establecerá la conexión con servidores centrales para mantener la actualización de los datos en ambos sentidos.

**RNF.05** El tiempo de entrenamiento requerido para que usuarios normales y avanzados sean productivos operando el sistema es de 15 días.

**RNF.06** Debe poseer una interfaz agradable para el cliente.

- **Requisitos de Fiabilidad**

**RNF.07** El sistema estará disponible 24 horas al día, 7 días a la semana.

**RNF.08** Disponibilidad de los casos asignados desde cualquier parte del país.

**RNF.09** El tiempo medio entre fallos es de 1 mes.

**RNF.10** El tiempo permitido para que el sistema quede fuera de operación luego de haber fallado es de 2 días.

**RNF.11** La precisión y exactitud requerida en las salidas del sistema o sea el máximo de errores, es de 5 errores/MLC.

**RNF.12** La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores.

- **Requisitos de Eficiencia**

**RNF.13** Tiempo de respuesta promedio de las peticiones que se realizan al servidor no deberá ser mayor de 3 segundos.

**RNF.14** El número de clientes o transacciones que el sistema puede alojar es de 2000.

- **Requisitos de Seguridad**

**RNF.15** Protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.

**RNF.16** El sistema debe mantener en todo momento la seguridad de la información asegurando la autenticidad de la misma.

**RNF.17** El sistema debe garantizar la confidencialidad, integridad y disponibilidad de la información que se procese en el sistema.

**RNF.18** La seguridad se establecerá por roles que se le asignarán a los usuarios que interactúen con el sistema.

**RNF.19** El software brindará solamente aquellas funcionalidades que competen a la Unidad Ejecutora donde esté implantado.

**RNF.20** El sistema mantendrá en todo momento las trazas que se corresponden con las diferentes situaciones críticas que puedan ocurrir.

- **Requisitos de Soporte**

**RNF.21** Soporte para grandes volúmenes de datos y velocidad de procesamiento.

**RNF.22** Tiempo de respuesta rápido en accesos concurrentes.

- **Requisitos de Restricciones de Diseño**

**RNF.23** El lenguaje de programación es PHP.V. 5.5.9.

**RNF.24** El framework de desarrollo es Sauxe 1.0.

**RNF.25** La herramienta IDE de desarrollo utilizada será NetBeans.

**RNF.26** La herramienta case utilizada es el Visual Paradigm para UML .V.6.1.

**RNF.27** La herramienta case utilizada para el modelado de la Base de Datos ER/Studio.V.7.1.

**RNF.28** Se utilizará el patrón de arquitectura Modelo – Vista – Controlador.

**RNF.29** La herramienta gestor de base de datos es el PostgreSQL 8.4

- **Requisitos de Interfaz**

**RNF.30** El sistema tiene que ofrecer una interfaz amigable, fácil de operar.

**RNF.31** El sistema tiene que mantener la línea de diseño establecida para la institución que mantiene la uniformidad y representatividad de la misma.

**RNF.32** Diseño sencillo, con pocas entradas, permitiendo que no sea necesario mucho entrenamiento para que los usuarios puedan utilizar el sistema.

- **Requisitos Legales y de Derecho de Autor**

**RNF.33** El Sistema de Gestión para los Tribunales Populares Cubanos tiene que 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.

- **Requisitos de Estándares Aplicables**

**RNF.34** 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 Symphony, específicamente el estándar “Upper Camel Case” para nombre de clases y variables.

### 2.4 Definición de Actores del Sistema

Después de haber obtenidos los requisitos del software se identificaron como actores del sistema:

Tabla: Descripción de Actores del Sistema.

Actor	Descripción
Secretaria de sala	Podrá realizar las siguientes acciones: <ul style="list-style-type: none"><li>• Registrar escrito.</li><li>• Modificar escrito promocional.</li><li>• Crear acta de comparecencia</li><li>• Registrar problemas a deliciar.</li><li>• Registrar escrito promocional que permite: registrar persona jurídica y registrar persona natural.</li></ul>
Abogado	Además de las acciones como usuario podrá notificarse de sus resoluciones y podrá realizar las siguientes acciones: <ul style="list-style-type: none"><li>• Registrar escrito promocional.</li></ul>

## Capítulo 2: Solución Propuesta

<b>Juez ponente</b>	<p>Es una generalización del actor presidente de sala. Podrá realizar las siguientes acciones:</p> <ul style="list-style-type: none"><li>• Crear, modificar y leer las siguientes resoluciones:<ul style="list-style-type: none"><li>○ Providencia de subsanación.</li><li>○ Providencia de nuevo señalamiento.</li><li>○ Providencia de AS.</li><li>○ Auto de rechazo de plano.</li><li>○ ANHLA.</li><li>○ Auto de desistimiento.</li><li>○ APAE.</li><li>○ ANPAE.</li><li>○ Auto de RS.</li><li>○ Auto declarando extemporáneo.</li><li>○ Providencia de archivo.</li></ul></li><li>• Visualizar los escritos presentados por el abogado.</li><li>• Visualizar los siguientes reportes propios:<ul style="list-style-type: none"><li>○ Expedientes fuera de término.</li><li>○ Expediente por monto.</li><li>○ Expediente resueltos.</li><li>○ Expediente en trámite.</li><li>○ Informe estadístico de Diligencias previas.</li></ul></li></ul>
<b>Presidente de sala</b>	<p>Es una generalización del actor Juez ponente podrá realizar todas las acciones del mismo y además</p> <ul style="list-style-type: none"><li>• Modificar la composición de tribunal</li></ul>
<b>Registrador de escrito</b>	<p>Es una generalización del actor Abogado y Secretaria de sala. Es el encargado de registrar:</p> <ul style="list-style-type: none"><li>• Escrito Promocional.</li><li>• Otros escritos.</li></ul>
<b>Usuario</b>	<p>Es una generalización de los actores: Registrador de escrito y Juez ponente, es el encargado de Visualizar los expedientes.</p>

## 2.5 Diagrama de Caso de Uso del Sistema

Los requisitos funcionales obtenidos del análisis del procedimiento Diligencias Previas, fueron agrupados en 18 casos de uso, inicializados por los 6 actores del sistema identificados anteriormente. Se aplicaron algunos de los patrones estudiados durante el Capítulo 1 con el objetivo de obtener el Diagrama de Casos de Uso del Sistema de forma que se cumpliese con las reglas descritas en anteriormente para la realización de este artefacto, dichos patrones son: Generalización Especialización entre actores y la Extensión concreta e Inclusión concreta entre casos de uso, además del patrón de Concordancia en su forma Reuso y Adición.

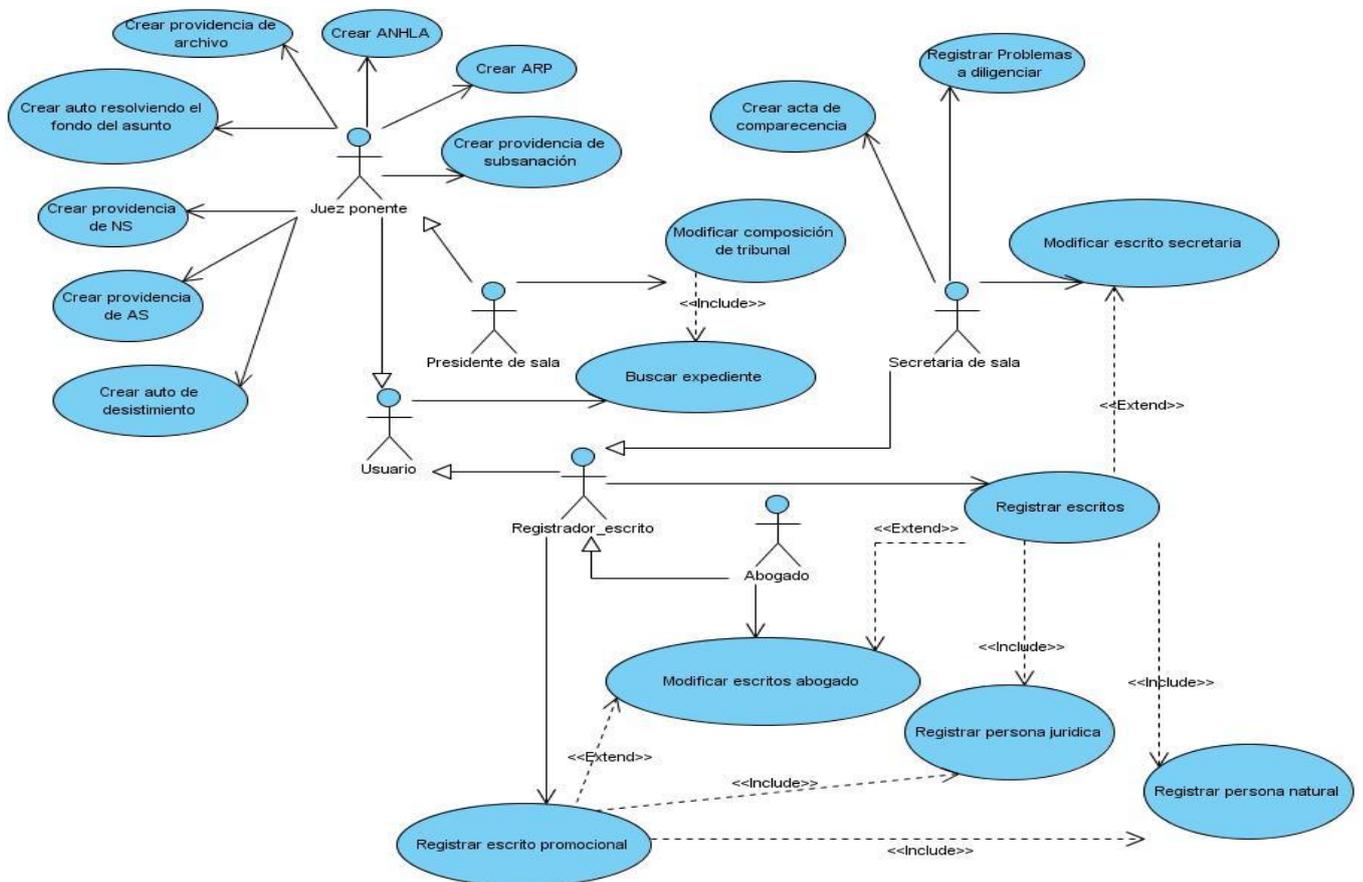


Figura 2.2 Diagrama de Casos de Uso del Sistema.

### 2.5.1 Realización de los Casos de Uso del Sistema

A continuación se muestra la descripción de uno de los Casos de Uso del Sistema con sus respectivos prototipos. Las restantes descripciones se pueden encontrar en el documento: **Modelo del Sistema Económico**, que se adjunta a la investigación.

Descripción Caso de Uso: Registrar Escritos

<b>Caso de Uso:</b>	Registrar escritos
<b>Actores:</b>	Registrador de escrito (inicia)
<b>Resumen:</b>	El caso de uso se inicia cuando el registrador de escrito accede al sistema para registrar los datos de los diferentes escritos, excepto el escrito promocional. Para ello introduce en el sistema los datos asociados a cada tipo de escrito, el sistema lo registra e informa al ponente del nuevo escrito presentado; de este modo termina el caso de uso.
<b>Precondiciones:</b>	El usuario debe estar autenticado correctamente en el sistema.
<b>Referencias</b>	RF.01, RF.02, RF.03, RF.08
<b>Reglas del negocio</b>	2, 3, 4, 6
<b>Prioridad</b>	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso comienza cuando el registrador de escrito accede al sistema para registrar los datos del escrito.	2. Solicita el número del expediente para el cual se presenta el escrito (solo si el registrador de escrito es secretaria).
3. Introduce el número del expediente. 4. Confirma la operación.	5. Muestra los datos principales del expediente y solicita: <ul style="list-style-type: none"> <li>• Tipo de escrito (obligatorio) <ul style="list-style-type: none"> <li>○ Evacuando término de subsanación</li> <li>○ Desistimiento.</li> <li>○ De súplica.</li> <li>○ Modificando pretensiones</li> </ul> </li> </ul>

## Capítulo 2: Solución Propuesta

---

	<ul style="list-style-type: none"><li>○ Otros</li><li>• Parte que presenta (obligatorio) solo si el registrador de escrito es secretaria.</li><li>• Argumentos del escrito (obligatorio) solo si el registrador de escrito es abogado.</li><li>• Documentos que acompañan (opcional).</li></ul>
<p>6. Introduce los datos.</p> <p>a) Si seleccionó <i>modificando pretensiones</i>, ver <b>sección 1</b>.</p> <p>7. Confirma la operación.</p>	<p>8. Valida que los datos introducidos son correctos, que los campos obligatorios no están vacíos y guarda los datos.</p> <p>9. Informa al ponente del nuevo escrito registrado.</p> <p>Termina el caso de uso.</p>

## Prototipo de Interfaz para la secretaria

The screenshot displays a web application interface for 'Tribunales Populares Cubanos'. The top navigation bar includes 'Materias', 'Buscar', 'Reportes', 'Calendario', and 'Administración y Gobierno'. The user is logged in as 'Bienvenido "Usuario"' on '14 de Julio de 2010'. The main content area is titled 'Inicio>Económico>'. On the left, there is a sidebar menu with categories: 'Expedientes a trabajar' (Citaciones, Pdte comparecencia, Términos vencidos), 'Registrar' (Escrito promocional, Notificación, Resulta, Otro escrito), 'Modificar' (Escrito, Otro escrito), 'Despachos recibidos (3)', and 'Despachos devueltos (2)'. The main area features a '+ Filtrar Búsqueda' section with a 'Ver Detalles' button (1). Below it is a table with columns 'Número', 'Trámite', and 'Fecha Venc.'. The table contains four rows of data, with checkboxes in the 'Número' column and yellow callout boxes (2, 3, 4, 5) next to the 'Fecha Venc.' values. A modal dialog titled 'Número de expediente' is open, showing a text input field with 'enter text...' and 'Aceptar' (6) and 'Cancelar' (7) buttons.

Número	Trámite	Fecha Venc.
<input type="checkbox"/> 1258*	Registrar resulta	28/09/2010
<input type="checkbox"/> 1345*	Crear acta de comparecencia	12/10/2010
<input type="checkbox"/> 1563*	Registrar escrito	25/10/2010
<input type="checkbox"/> 1563*		X

**Número de expediente**

enter text...

Aceptar Cancelar

# Capítulo 2: Solución Propuesta

**Tribunales Populares Cubanos**

Materias | Buscar | Reportes | Calendario | Administración y Gobierno

Inicio>Económico> Bienvenido "Usuario"  
14 de Julio de 2010

- Expedientes a trabajar
  - Citaciones
  - Pdte comparecencia
  - Términos vencidos
- Registrar
  - Escrito promocional
  - Notificación
  - Resulta
  - Otro escrito
- Modificar
  - Escrito
  - Otro escrito
- Despachos recibidos (3)
- Despachos devueltos (2)

**Datos primarios**

Número del expediente: 102011  
Proceso: Diligencias previas  
Causal: Confesión de deudas [Ver más detalles](#) <sup>4</sup>

Tipo de escrito:  <sup>3</sup>      Parte:  <sup>5</sup>

**Escrito**

**Documentos que acompañan**

Ruta:

<sup>1</sup>       <sup>2</sup>

COPYRIGHT © 2010, Tribunales Populares Cubanos

## Prototipo de Interfaz para el abogado

The screenshot displays a web application interface for 'Tribunales Populares Cubanos'. The interface is divided into several sections:

- Header:** Includes a logo, the text 'Tribunales Populares Cubanos', a search bar labeled 'Buscar', and a user greeting 'Bienvenido "Usuario" 14 de Julio de 2010'.
- Navigation:** A sidebar on the left contains a menu with 'Inicio>Económico' and two main categories: 'Mis expedientes' (with sub-items 'Expedientes en trámites' and 'Notificaciones') and 'Presentar escritos' (with sub-items 'Escrito inicial', 'Personería', and 'Conclusiones provisionales').
- Datos primarios:** A section containing case information: 'Número del expediente: 102011', 'Proceso: Diligencias previas', and 'Causal: Confesión de deudas'. A link 'Ver más detalles' with a small yellow icon is present.
- Tipo de escrito:** A dropdown menu currently showing '-Seleccione-' with a small yellow icon.
- Escrito:** A large, empty text area for entering the document content.
- Documentos que acompañan:** A section with a 'Ruta:' input field, an 'Examinar' button, and a '+ Nuevo' button.
- Buttons:** At the bottom right, there are 'Enviar' and 'Cancelar' buttons, each with a small yellow icon.

COPYRIGHT © 2010, Tribunales Populares Cubanos

## Capítulo 2: Solución Propuesta

Flujos Alternos	
<b>Flujo Alternativo 1 “Insertar datos incorrectos y/o campos vacíos”</b>	
Acción del Actor	Respuesta del Sistema
3, 5. Introduce datos incorrectos.	3.1 Informa que los datos son incorrectos y no guarda los cambios.
<b>Flujo Alternativo 2 “Cancelar operación”</b>	
4, 6. Cancela la operación	4.1 Cierra la interfaz y no guarda los cambios.
<b>Sección 1 “Registrar escrito modificando pretensiones”</b>	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción <i>modificando pretensiones</i> .	2. Solicita los siguientes datos: <ul style="list-style-type: none"> <li>• Causal:               <ul style="list-style-type: none"> <li>○ reconocimiento de firma</li> <li>○ confesión de deuda</li> <li>○ ambos</li> </ul> </li> <li>• Datos de la deuda:               <ul style="list-style-type: none"> <li>○ monto</li> <li>○ moneda</li> <li>○ sucursal</li> <li>○ número de cuenta</li> </ul> </li> </ul> Continúa el flujo normal de los eventos en el paso7.
<b>Poscondiciones</b>	<ul style="list-style-type: none"> <li>• Si se registró el escrito evacuando término de subsanación, en el término concedido, el estado del expediente cambió a “pendiente de admisión”.</li> <li>• Si se registró el escrito evacuando término de subsanación fuera de término, el estado del expediente cambió a “pendiente de fallo” y el estado del escrito cambió a “fuera de término”.</li> <li>• Si se registró el escrito de desistimiento, el estado del expediente cambió a “pendiente de fallo”.</li> <li>• Si se registró el escrito de súplica en término, el estado del expediente cambió a “recurrido en súplica”.</li> <li>• Si se registró el escrito de súplica fuera de término, el estado del escrito cambió a “fuera de término” y el expediente se mantuvo en su estado actual.</li> <li>• Se registró la acción realizada.</li> </ul>

## 2.6 Conclusiones Parciales

La realización del modelo de negocio permite entender mucho mejor la estructura y los procesos que se llevan a cabo en la Sala de lo Económico de los Tribunales Populares Cubanos y constituye la entrada fundamental del siguiente flujo de trabajo propuesto por la metodología RUP: levantamiento de requisitos.

Mediante la realización del levantamiento de requisitos se identificaron los requisitos funcionales y los no funcionales del sistema para conocer las funcionalidades que debe brindar el Módulo Económico y las restricciones sobre las que va a operar el sistema, siempre cumpliendo con las necesidades y posibilidades del cliente. Se identificaron y describieron los actores y casos de uso del sistema.

## CAPÍTULO 3: ANÁLISIS DE LOS RESULTADOS

### 3.1 Introducción

En el presente capítulo se definen las métricas que se aplicarán para la validación de los artefactos generados en el capítulo anterior: Requisitos funcionales y los Requisitos no Funcionales así como el Diagrama de Casos de Usos del Sistema. A los requisitos obtenidos se aplica la métrica para determinar la especificidad de los requisitos y en el DCUS se aplica la métrica para verificar la calidad de la funcionalidad del diagrama de casos de uso de sistema, también se refleja el acta de liberación de los artefactos generados por los especialistas de la Materia Económica.

### 3.2 Métrica para la calidad de especificación de requisitos de software

Para validar los artefactos generados durante la fase de análisis se aplicaron métricas de medición a cada uno de ellos, el equipo de revisores lo conformaron 4 especialistas principales de la materia, dos analistas y 2 programadores, en el presente epígrafe se refleja la métrica aplicada a los requisitos obtenidos y en el siguiente epígrafe se refleja la métrica aplicada a el DCUS y los resultados obtenidos por cada uno de ellos.

A los requisitos de software obtenidos se aplicó la métrica para medir la calidad de especificidad de los mismos, estudiada en el Capítulo 1, en la siguiente tabla se refleja los resultados obtenidos:

Especificidad				
Tipo de requisito		Interpretaciones		Total
		Iguales	Desiguales	
Funcionales	51	47	4	51
No funcionales	34	34	0	34
Total	85	81	4	85

$$RF= 51$$

$$RNF=34$$

$$NR=RF+ RNF= 51 + 34 = 85$$

$$Q1=NRI/ NR=81 /85 = 0.95$$

## Capítulo 3: Análisis de los resultados

El valor resultante de Q1 igual a 0.95, valor muy aproximado a 1 demuestra que el grado de ambigüedad en la especificación de los requisitos de software del Módulo Económico fue muy bajo y por consiguiente se comprobó calidad en la especificación. Con el margen de error existente, con valor igual a 0.5 se procedió a una revisión de los requisitos que presentaban dificultades, por parte del equipo de analistas a revisar estos requisitos y darle solución. Después de la revisión se volvió a aplicar esta métrica, concluyendo que todos los requisitos tuvieron la misma interpretación y que los requisitos obtenidos se encuentran con un alto grado de especificidad.

### 3.3 Métricas para la funcionalidad del diagrama de casos de uso del sistema

Seguidamente se muestra una tabla conformada por 4 factores: Completitud, Consistencia, Correctitud y Complejidad, con el objetivo de medir la calidad de la funcionalidad del DCUS, donde se especifican las métricas asociadas a cada factor, con sus umbrales (valor de aceptación) correspondientes y se le proporcionará el valor arrojado a cada una de ellas en las 2 iteraciones realizadas por los revisores.

**Tabla 3:** Factores en la métrica para la funcionalidad del diagrama de casos de uso.

Factores	Métricas asociadas	Valor (%)	
		Iteración 1	Iteración 2
<b>Completitud</b>			
1. ¿Han sido involucradas todas las áreas funcionales relevantes a las cuales apoyará el sistema?	1. Número de áreas funcionales relevantes omitidas. Umbral: < 10% Acciones sugeridas: Revisar los requisitos del negocio.	0%	0%
2. ¿Han sido definidos todos los roles relevantes de usuario encargados de generar/modificar o consultar información?	2. Número de roles relevantes omitidos. Umbral: < 10% Acciones sugeridas: Revisar el alcance del sistema e involucrar tipos de usuarios representativos de cada una de las áreas funcionales.	16% (omitidos 1 de 6) Se omitió el rol de Registrador de escrito.	0%

## Capítulo 3: Análisis de los resultados

<p>3. ¿Están definidos todos los requisitos que justifican la funcionalidad del caso de uso?</p>	<p>3. Número de requisitos omitidos por caso de uso. Umbral &lt; 10% Acción sugerida: Revisar la lista de requisitos para determinar cuáles serán apoyados por cada caso de uso.</p>	<p>0%</p>	<p>0%</p>
<p>4. ¿Existen requisitos que no han sido considerados en algún caso de uso?</p>	<p>4. Número de requisitos que no son considerados en ningún caso de uso. Umbral &lt; 10% Acción sugerida: Revisar la lista de requisitos para determinar cuáles deben ser incorporados en algún caso de uso.</p>	<p>1.96% (omitidos 1 de 51) RF.04 Asignar número de expediente.</p>	<p>0%</p>
<p>5. ¿Están todas las acciones del flujo de eventos redactadas en función del responsable?</p>	<p>5. Número de acciones del flujo de eventos que no están redactadas en función del responsable. Umbral &lt; 20% Acción sugerida: Revisar las responsabilidades tanto del actor (actores) como del sistema.</p>	<p>0%</p>	<p>0%</p>
<b>Consistencia</b>			
<p>6. ¿El nombre dado a los casos de uso es una expresión verbal que describe alguna funcionalidad relevante en el contexto del usuario?</p>	<p>6. Número de casos de uso que tienen un nombre incorrecto. Umbral &lt; 20% Acción sugerida: Modificar el nombre del caso de uso de tal manera que signifique una acción desde el punto de vista del usuario.</p>	<p>5.55% (mal redactado 1 de 18) CU: Registrar escrito, sería (escritos)</p>	<p>0%</p>

## Capítulo 3: Análisis de los resultados

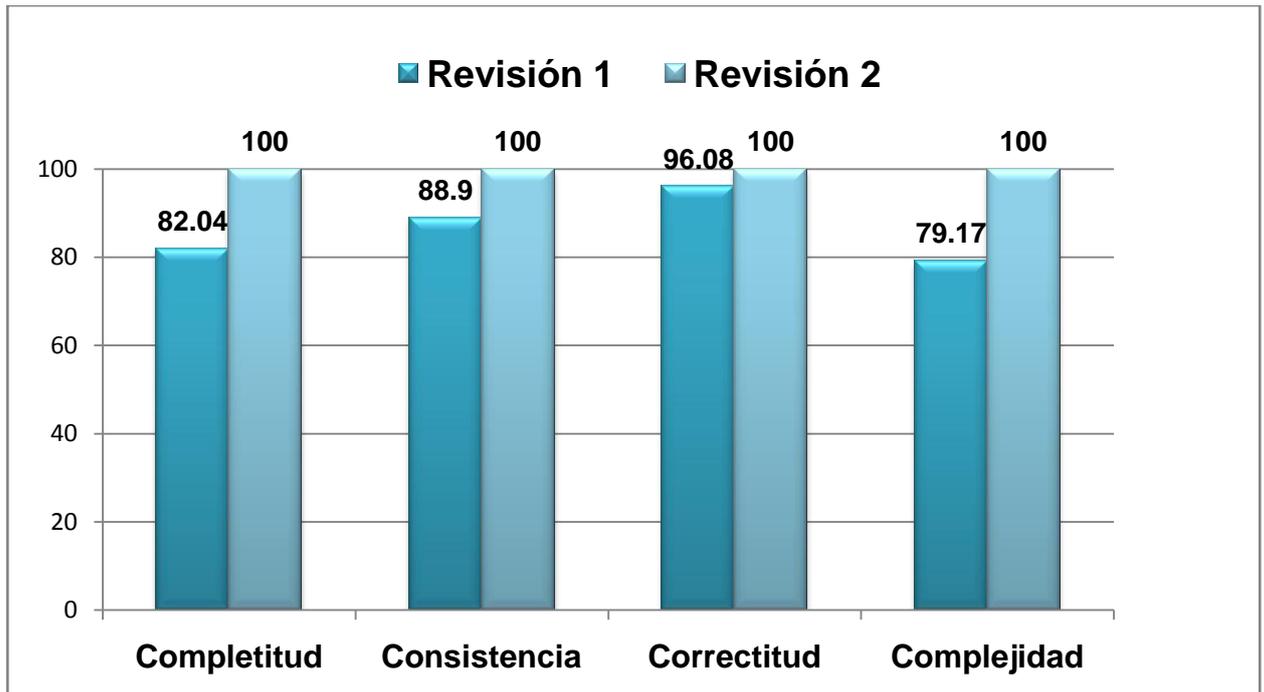
<p>7. ¿Representa el caso de uso una interacción observable por un actor?</p>	<p>7. Número de casos de uso que no representan una interacción observable por un actor.</p> <p>Umbral &lt; 5%</p> <p>Acción sugerida: Eliminar el caso de uso e incorporar su funcionalidad como una responsabilidad del sistema dentro de otro caso de uso.</p>	<p>0%</p>	<p>0%</p>
<p>8. ¿Está adecuadamente redactado (en el lenguaje del usuario) el flujo de eventos?</p>	<p>8. Número de casos de uso no aceptados.</p> <p>Umbral &lt; 10%</p> <p>Acción sugerida: Corregir la descripción del caso de uso.</p>	<p>5.55% (omitidos 1 de 18)</p> <p>Problemas en la descripción de el CU: Registrar escrito promocional</p>	<p>0%</p>
<b>Correctitud</b>			
<p>9. ¿Representa el caso de uso requisitos comprensibles por el usuario?</p>	<p>9. Número de requisitos que no son comprensibles por el usuario</p> <p>Umbral: &lt; 5%</p> <p>Acción sugerida: Discutir con el usuario la interacción que describe el caso de uso y ajustar dicha descripción de manera que sea comprensible por el usuario.</p>	<p>3.92% (omitidos 2 de 51)</p> <p>Los requisitos fueron: RF.42 Visualizar acciones, RF.44 Informar de escrito presentado.</p>	<p>0%</p>

## Capítulo 3: Análisis de los resultados

<p>10. ¿Existe para cada caso de uso por lo menos un usuario responsable?</p>	<p>10. Número de casos de uso que no tienen un usuario responsable. Umbral: &lt; 10% Acción sugerida: Analizar la responsabilidad que representa el caso de uso y acordar con los usuarios cuál es el responsable directo de éste.</p>	<p>0%</p>	<p>0%</p>
<b>Complejidad</b>			
<p>11. ¿Los elementos dentro del diagrama están adecuadamente ubicados de manera que facilitan su interpretación?</p>	<p>11. Número de elementos del diagrama que requieren reubicación. Umbral: &lt; 30% Acción sugerida: Modificar la ubicación de los elementos del diagrama de manera que los elementos relacionados se encuentren lo más cercano posible.</p>	<p>20.83% (omitidos 5 de 24) Los actores: Usuario, Registrador de escrito y Abogado. Los CU: Registrar escritos y CU: Modificar escrito abogado</p>	<p>0%</p>

Los resultados obtenidos muestran el valor arrojado por el equipo de revisión provocado por la aplicación de la métrica para la funcionalidad del DCUS. Se realizó una primera revisión donde se detectaron algunas no conformidades, el valor definido por el umbral de cada una de las métricas asociadas a cada factor nunca es superado, por lo que estas no conformidades no son críticas ni representan grandes amenazas para el modelado del DCUS. Se corrigieron todas las no conformidades hasta conseguir en la segunda iteración un diagrama de casos de uso del sistema con 100% de calidad de su funcionalidad, teniendo en cuenta los atributos medibles descritos en la descripción de la métrica, como se refleja en la Gráfica 3.1.

## Capítulo 3: Análisis de los resultados



Gráfica 3.1: Atributos para medir calidad de la funcionalidad del DCUS.

# Capítulo 3: Análisis de los resultados

## 3.4 Acta de aceptación de los requisitos

Una vez medidos los requisitos del sistema y después de haber corregido las deficiencias encontradas se procede a la firma del acta de aceptación por parte de los clientes:



### Acta de Conformidad de Artefactos, Tribunal Supremo Popular (TSP).

Martes, 31 de mayo de 2011

Año 53 de la Revolución

Luego de haber efectuado una serie de revisiones con los jueces de la Sala de lo Económico del TSP, así como varias ediciones de cursos de capacitación, con la presencia de Presidentes de las Salas de lo Económico en Tribunales Provinciales de todo el país, revisando detalladamente los artefactos: Modelo de negocio, Especificación de Requisitos y Modelo de Sistema, pertenecientes al módulo Económico del Sistema de Informatización de Tribunales, del Centro GEGEL de la Facultad 3. Se puede afirmar que han corregido completamente los elementos señalados por parte de los especialistas. En general existe una conformidad total con la documentación presentada.

Asesora legal y juez del Tribunal supremo popular  
Marly Fuentes águila

Asesora legal y Juez del Tribunal Provincial Popular de la Habana  
MSc. Carmen castillo

## 3.5 Conclusiones Parciales

Una vez determinados los Requisitos Funcionales y los No Funcionales y realizado el DCUS se definieron un conjunto de métricas a tener en cuenta para la validación de dichos artefactos, mediante las cuales se pudo comprobar que los requisitos del sistema carecían de ambigüedad aplicando la métrica de la comprobación de la calidad en dicha especificación. También se aplicaron métricas para la funcionalidad del DCUS demostrando su buena construcción. Se confeccionó el acta de aceptación por parte de los clientes demostrando que la especificación de requisitos y el DCUS cubren sus expectativas y necesidades.

## CONCLUSIONES

Con la realización de este trabajo se arriba a los siguientes resultados:

- El estudio del proceso de desarrollo de software y la ingeniería de requisitos, junto a las metodologías, lenguajes de modelado, patrones de casos de uso, las métricas de validación y herramientas, permitió conocer cuáles eran las que más se adaptaban al desarrollo del procedimiento Diligencias Previas y condujeron al modelado del sistema, teniendo en cuenta las necesidades del cliente.
- El análisis del procedimiento Diligencias Previas, así como las entrevistas y reuniones realizadas en conjunto con los especialistas, permitió determinar las actividades a informatizar, además de los actores que interactúan con las diferentes funcionalidades del sistema.
- Se obtuvieron y especificaron los requisitos de software a través de la modelación del sistema, los cuales fueron validados mediante la aplicación de diferentes técnicas y métricas de calidad de software, demostrando que los requisitos especificados no son ambiguos y que el DCUS cubre todas las necesidades de los clientes.

## RECOMENDACIONES

- Dar continuidad a los flujos de trabajo propuestos por RUP a partir de los casos de uso descritos en el presente trabajo, logrando de esta forma una solución informática que sea capaz de resolver los problemas reales para los cuales fue creada.
- Realizar la gestión de requisitos para tener un control sobre los riesgos que puedan atentar contra el buen desempeño del software, ya que este es un negocio muy cambiante debido a todas las transformaciones que constantemente en él se están llevando a cabo.

## BIBLIOGRAFÍA

1. Eldia.es. [Online] 10 19, 2010. [Cited: 01 26, 2011.] <http://www.eldia.es/2010-10-19/canarias/39-PSC-censura-Atlante-II-lleve-dos-anos-funcionar-juzgados-Canarias.htm>.
2. Justizia.net. [Online] [Cited: 01 12, 2011.] <http://www.justizia.net/docuteca/Documentos/1381Justicia%20Civil%201%20Instancia%20PafsVasco.PDF>.
3. ripj. [Online] [Cited: 01 12, 2011.] [http://www.ripj.com/art\\_jcos/art\\_jcos/num23/LexNet.pdf](http://www.ripj.com/art_jcos/art_jcos/num23/LexNet.pdf).
4. Agile Software Development. Highsmith Series. [book auth.] Alistair Cockburn.
5. **Gary, Chin.** Agile Project Management: How to Succeed in the Face of Changing Project Requirements. 2004.
6. informatizate.net. [Online] [Cited: 01 17, 2011.] [http://www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html).
7. **Pressman, Roger S.** *Ingeniería de Software. Un enfoque práctico.* 1997.
8. **Smith, John.** A comparison of RUP and XP. [Online] 2001. [Cited: 01 18, 2011.] <ftp://ftp.software.ibm.com/software/rational/web/whitepapers/2003/TP167.pdf>.
9. **Kruchten, Philippe.** *The Rational Unified Process An Introduction.* Marzo 2000. ISBN 0-201-70710-1..
10. **Kroll, Per y Kruchten, Philippe.** *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP.* . Abril 2003. ISBN 0-321-16609-4. .
11. **Schwaber, Ken.** *Agile Software Development with Scrum.*
12. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software.* Madrid : s.n., 2000. ISBN 84-7829-036-2.
13. **A, Díez.** *IRqA y el desarrollo de proyectos: Experiencias Prácticas. I Jornadas de Ingeniería de Requisitos.* 2001.
14. **Durán A, Bernárdez B, Toro M.** *A Requirements Elicitation Approach Based in Templates and Patterns. Workshop de Engenharia de Requisitos.* . Buenos Aires : s.n., 1999.
15. **Pan, D., Zhu, D., Johnson, K.** *Requirements Engineering Techniques. Internal Report. Department of Computer Science.* University of Calgary : s.n., 2001.

16. **IBM OOTC.** *Developing Object Oriented Software.* IBM Object Oriented Technology Center. Prentice-Hall. 1997.
17. **Raghavan, S., Zelesnik, Ford, G.** *Lectures Notes of Requirements Elicitation.* 1994. CMU/SEI-94-EM-10.
18. **Vilain, P., Schwabe, D., Sieckenius, C.** *A diagrammatic Tool for Representing User Interaction in UML.* York : s.n., 2000.
19. Aiteco. [Online] [Cited: 01 29, 2011.] <http://www.aiteco.com/modelpro.htm>.
20. **Davis, R.** *Business Process Modeling with ARIS: A Practical Guide.* 4ta Edition 2005. 2011. p. 531 p.
21. **S. F. Howard Peter.** *Business Process Management: The Third Wave.* 2003.
22. **Kaisha-Tec 2008.** Comfortable and Intuitive Process Modeling and BPM Design. [Online] Marzo 15, 2008. <http://www.activemodeler.com/AvantageFoundation>.
23. **White, S. A.** *Business Process Modeling Notation.* 2003. p. 189.
24. **Scheer, I.** 'ARIS Method'. 2004. p. 2109.
25. **Visual Paradigm.** UML CASE Tools - Free for Learning UML, Cost-Effective for Business Solutions. [Online] [Cited: 01 27, 2011.] <http://www.visual-paradigm.com/product/vpuml/>.
26. **Microsoft 2007.** Datos básicos sobre Office Visio 2007. [Online] Febrero 12, 2008. [Cited: 01 28, 2011.] <http://www.microsoft.com/latam/office/visio/prodinfo/facts.mspx>.
27. DoRol. [Online] 2007. [Cited: 01 26, 2011.] <http://dorol.wordpress.com>.
28. **Axure.** *Axure.* [Online] 2007. [Cited: 01 24, 2011.] <http://axure-rp.softonic.com>.
29. **Gunnar Overgaard, Karin Palmvist, Addison-Wesley.** *Use Cases- Patterns and Blueprints.* 2005.
30. **Frank Armour, Granville Miller.** *Advanced Use Case Modeling.* 2001.
31. **Fowler, Martin.** *Analysis Patterns.* 1997.
32. **S.H.Kan.** *Metrics and Models in software Quality Engineering.* s.l. : Addison – Wesley, 1995.
33. **Márquez, José Manuel.** Docente de la Universidad de Valladolid. : s.n., 2009.
34. **José Manuel Márquez, Docente de la Universidad de Valladolid.**
35. **José Manuel Márquez.** *Ingeneria de Requisitos.* Docente de la Universidad de Valladolid. : s.n.
36. **Iruetagoiena, Alicia.** *Curso de presentación oral UNIR.*

37. **IEEE**. [Online] IEEE Standard Glossary of Software Engineering Terminology. [Cited: 01 21, 2011.] <http://standards.ieee.org/reading/ieee/std/se/610.12-1990.pdf>. IEEE Standard 610.12-1990.
38. **beck, Kent**. *Extreme Programming Explained*. Addison Wesley. Septiembre 1999. ISBN 0201616416. .
39. **umbrello-uml-modeller**. Umbrello. [Online] <http://uml.sourceforge.net/>.
40. kde. [Online] [Cited: 01 27, 2011.] <http://docs.kde.org/stable/es/kdesdk/umbrello/index.html> .
41. **EducaMarketing**. *Guía para realizar presentaciones en público*. 2005.

## GLOSARIO DE TÉRMINOS

**Auto:** Resolución judicial dictado en forma dirigida a decidir cuestiones esenciales no referidas a la solución del litigio.

**Auto de Rechazo de Plano:** Es un tipo de auto dictado por el juez cuando el asunto que se presenta ante el tribunal no es de su jurisdicción o competencia.

**Auto de Inadmisión (ANHLA):** A pesar que el asunto a conocer es de la jurisdicción y competencia del tribunal, se decide no admitir el escrito promocional, dado que existen errores formales en los mismos que no se han subsanado, o que falta cierto requisito esencial para poder abrirse el proceso, como ejemplo que no sea la persona facultada para demandar o para ser demandada.

**Cédula de citación:** Es el medio formal por el que se realiza la diligencia de citación.

**Carpeta de CEA:** Carpeta de control de expedientes archivados.

**Radicar:** Es la acción donde la secretaria registra por el número consecutivo los expedientes para su tramitación en el Libro de Radicación de Escritos.

**Diligencia doy cuenta:** Diligencia por medio de la cual el secretario informa sobre el trámite y provoca que el tribunal emita la resolución correspondiente.

**DP:** Diligencias Previas al módulo Ejecutivo.

**Desistimiento AC:** Desistimiento antes de la citación ejecutiva.

**Desistimiento DC:** Desistimiento después de la citación ejecutiva.

**Expediente:** Es un legajo que contiene toda la tramitación judicial en un caso concreto que se enumera consecutivamente por año.

**Escritos promocionales:** Escrito que da inicio a la tramitación judicial.

**Escritos de personería:** Escrito por el cual se formaliza la intervención en la tramitación judicial.

**Escritos conciliatorios:** Escrito en el que se consigna el acuerdo entre las partes sobre las cuestiones en conflicto.

**Libro de presentación de escritos (LPE):** Es un libro registro en el que se deja constancia oficial de la presentación de escritos y documentos acompañados.

**Libro de Radicación de escritos (LRE):** Es el libro en el que se registra por su número consecutivo los expedientes para su tramitación.

**Libro de numeración de auto (LNA):** Existen autos que ponen fin a los procesos o a trámites como la diligencia previa (las diligencias previas como no son procesos no terminan mediante sentencia, siempre lo harán por auto), y estos se asientan en este libro en orden consecutivo, y es este el que les da número. (Ejemplo Auto No. 67/09).

**Providencia:** Resolución judicial destinada al impulso de la tramitación procesal.

**Providencia de radicación:** Es la resolución por la que se dispone la formación del expediente. Su registro en el libro de radicación consignándose el número de orden correspondiente de año de tramitación y se dispone el turnado del expediente.

**Providencia de admisión:** Es la providencia por la que se dispone la tramitación del proceso y se ordena el trámite correspondiente.

**Providencia de subsanación:** Es la providencia que crea el juez ponente para mandar a subsanar los errores que existan en los escritos presentados.

**Resoluciones judiciales:** Son dictadas por los jueces. Estos pueden dictar tres tipos: providencia, auto y sentencia. Toda resolución debe ser notificada a las partes del proceso por la secretaria.

**Resulta:** Documento que el alguacil recibe al entregar la citación por parte de las partes.

**Retornar expediente:** Es la acción que realiza el presidente de la sala cuando el juez turnado no puede realizar el trámite. Sólo lo puede efectuar el presidente de la sala.

**Tablilla:** Es el medio de publicidad que normalmente produce el efecto de la notificación y se concretan otras diligencias.

**LPE:** Libro Presentación de Escritos.

**LR:** Libro de Radicación.

**LSD:** Libro de Salidas de Documentos.

**LNA:** Libro de Numeración de Autos.

**CEA:** Control de expedientes archivados.

**Desistimiento DC:** Desistimiento después de la citación.

**Desistimiento AC:** Desistimiento antes de la citación.

**CSD:** Cuaderno de salida de documentos.

**RS:** Recurso Súplica.

**PE:** presentación de escrito

**ANHLA:** Auto No Haber Lugar a Admitir

**APAE:** Auto Preparando la Acción Ejecutiva

**ANPAE:** Auto No Preparando la Acción Ejecutiva

**AS:** Admisión y Señalamiento

**NS:** Nuevo Señalamiento.