



**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**  
**Facultad Regional de Artemisa**

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

**Título:** Servidor para el módulo de la Dirección  
de Importaciones del Ministerio del Comercio  
Exterior y la Inversión Extranjera (MINCEX).

**Autor:** Yelina Rodríguez Pérez

**Tutor:** Ing. Annies Cedeño López

**Co-Tutor:** MSc. Rodolfo del C. Piedra Cabrera

**Artemisa, Junio 2012**

**“Año 54 de la Revolución”**

## Declaración de autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

\_\_\_\_\_  
Yelina Rodríguez Pérez  
Autor

\_\_\_\_\_  
Annies Cedeño López  
Tutor

\_\_\_\_\_  
Rodolfo del C. Piedra  
Co-Tutor

## *Frase*



*Incluso un camino sinuoso, difícil, nos puede conducir a la meta si no lo abandonamos hasta el final.*

*Paulo Coelho*

## Agradecimientos

*A mi mamá, por apoyarme siempre sin importar las consecuencias.*

*A mi papá, por confiar en mí y en las decisiones que he tomado.*

*A mis hermanas Yisel y Yaquelin, por estar siempre cuando más las he necesitado.*

*A toda mi familia, que es el sustento de mi vida.*

*A mi amiga Yarisel, que ha estado todo el camino y me ha enseñado que es la amistad.*

*A mi amigo Yosney, que sin importar la distancia, siempre ha tenido una frase reconfortante que decirme. Gracias.*

*A mi compañera de tesis Cristina, por ser quien es durante estos 5 años y por todas las noches de programación, que no fueron pocas.*

*A mis amigos, que son los que realmente me han soportado y apoyado durante cinco años, sin ellos no sería quien soy hoy.*

*Y finalmente a mi tutora Annies por todo el apoyo que me ha dado.*

*A todos, muchísimas gracias.*

## *Dedicatoria*

*A mis padres, por ser mi apoyo y la luz que me guía siempre.*

*A mis hermanas, que son lo mejor que me ha pasado en la vida.*

*A mi tía y mis abuelas, que aunque no pudieron verme realizar mi sueño, están presentes en todo lo que hago. Esto es para ellas.*

## Resumen

El Ministerio del Comercio Exterior y la Inversión Extranjera (Mincex) tiene como objetivo y misión esencial la de preparar y proponer al Gobierno la política integral del Estado y del Gobierno en cuanto a las actividades del comercio exterior. Este ministerio requiere de la automatización de sus procesos para mejorar la confiabilidad y rapidez de los mismos. Se analizaron, para la realización del sistema, los conceptos fundamentales asociados al problema y se estudiaron las herramientas y tecnologías más apropiadas para ello. Definiéndose de esta forma como Entorno de Desarrollo Integrado (IDE) NetBeans, que hace uso del lenguaje de programación Java y del framework jWebSocket. Se escogió como metodología SXP, definiéndose dentro de la misma, historias de usuario y tareas de ingeniería para una mejor comprensión de las funcionalidades a implementar y diagramas que ayudaron a un mejor entendimiento del negocio y a diseñar la mejor solución para el sistema.

**Palabras Claves:** Automatización, framework, Jwebsocket, metodología, SXP, funcionalidades, diagramas.

# Índice

INTRODUCCIÓN .....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA, TECNOLOGÍAS, HERRAMIENTAS Y METODOLOGÍAS DE DESARROLLO UTILIZADAS. ....	7
1.1 REFERENTES TEÓRICOS .....	7
1.2 SISTEMAS DE GESTIÓN DE INFORMACIÓN PARA EL COMERCIO EXTERIOR EN EL MUNDO.....	10
1.3 METODOLOGÍA, HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS .....	12
PARA EL DESARROLLO DE LA SOLUCIÓN PROPUESTA. ....	12
1.3.1 Metodología de desarrollo.....	12
1.3.2 Lenguaje de Modelado y Herramienta CASE.....	14
1.3.3 Entorno de Desarrollo Integrado.....	16
1.3.4 Lenguajes de programación.....	17
1.3.5 Herramientas de control de versiones .....	18
1.3.6 Frameworks.....	19
CAPÍTULO 2. CARACTERÍSTICAS, ANÁLISIS Y DISEÑO DEL SISTEMA.....	23
2.1 SOLUCIÓN PROPUESTA .....	23
2.2 INFORMACIÓN A AUTOMATIZAR .....	23
2.3 ARQUITECTURA Y PATRONES DE DISEÑO. ....	24
2.4 PLANIFICACIÓN DEL PROYECTO POR ROLES.....	26
2.5 MODELO DE DOMINIO.....	28
2.6 LISTA DE RESERVA DEL PRODUCTO (LRP).....	29
2.7 HISTORIAS DE USUARIOS Y TAREAS DE INGENIERÍA. ....	34
2.8 PLAN DE RELEASE.....	43
2.9 DISEÑO CON METÁFORA. ....	44
2.10 DIAGRAMA DE COMPONENTES.....	46
CAPÍTULO 3 IMPLEMENTACIÓN Y VALIDACIÓN DEL SISTEMA .....	49
3.1 ESTÁNDAR DE CÓDIGO .....	49
3.2 IMPLEMENTACIÓN DEL LADO DEL SERVIDOR.....	49
3.2.1 Eventos de la aplicación.....	49
3.2.2 Clases services de la aplicación.....	51
3.2.3 Clases plugin de la aplicación. ....	52
3.3 PRUEBAS AL CÓDIGO.....	54
3.3.1 Pruebas de caja Blanca .....	55
3.3.2 Resultados de las pruebas.....	61
3.4 RESULTADOS OBTENIDOS .....	61
3.5 FUNCIONALIDADES OBTENIDAS.....	61
3.6 APORTE SOCIAL Y ECONÓMICO .....	62
CONCLUSIONES GENERALES .....	63
REFERENCIAS BIBLIOGRÁFICAS .....	64
BIBLIOGRAFÍA.....	67

## Índice de figuras

Figura 1: Modelo de dominio .....	29
Figura 2: Diseño con metáfora.....	45
Figura 3: Diagrama de componentes.....	47
Figura 4: Evento addApproval .....	50
Figura 5: Plugin ApprovalPlugin.....	54

## Índice de tablas

Tabla 1: Planificación del proyecto por roles.....	26
Tabla 2: Lista de reserva del producto .....	30
Tabla 3: Historias de usuario y Tareas de la ingeniería .....	35
Tabla 4: Plan de release .....	44
Tabla 5: Métodos implementados en la clase ApprovalService .....	51
Tabla 6: Métodos implementados en la clase ApprovalPlugin .....	52
Tabla 7: Resultados obtenidos en las pruebas.....	61

## Introducción

Desde el surgimiento del hombre sobre la faz de la tierra este sintió la necesidad de comunicarse y de contar su historia. Para que esta llegase hasta nuestros días lo hizo siempre a través del almacenamiento de información, ya fuesen señales, escrituras de libros u otros medios, pero de lo que sí no hay duda es que ello ha servido para el desarrollo continuo de la humanidad.

Los inicios de la información se remontan al surgimiento del hombre, cuando este hacía pictografías en las cuevas, mucho antes de surgir un lenguaje que les permitiera comunicarse. Ya en la edad media el almacenamiento, acceso y uso de la información se limitaba a realizarse en las bibliotecas de los monasterios. Más tarde, con el nacimiento de la imprenta, los libros comienzan a fabricarse en serie. Surgen los primeros periódicos. Más tarde la radio, después la televisión y por último hasta ahora, la computadora e internet. Inventos estos que cambiarían para siempre el mundo de la comunicación y la información, dando inicio a una nueva etapa: la de las tecnologías de la información y la comunicación (TIC).

Las TIC siguen con un desarrollo vertiginoso, en ascenso. Ellas están presentes en una sociedad regida mayoritariamente por la cantidad de conocimientos que se pueden adquirir, donde los rápidos cambios, el aumento de los mismos y una educación de alto nivel, se convierten en una exigencia permanente. Estas han permitido llevar el mundo de la comunicación a un nivel diferente, facilitando la conexión entre las personas e instituciones de todo el mundo, eliminando barreras de espacio y tiempo. Siendo sus principales aportaciones a las actividades diarias, una serie de funciones que facilitan la realización de disímiles trabajos ya que siempre se requiere de cierta información para llevarlo a cabo.

Esta autora considera importante decir que la información es un recurso valioso que debe estar accesible para todos los usuarios. De ahí la necesidad de procesamiento de toda la información que se genera a diario en el mundo, que

condujo al surgimiento del término de gestión de la información a mediados de los años 70, cuando los sistemas informáticos empezaron a ser comunes en las organizaciones.

Es importante destacar que todo ello es el resultado de la necesidad que existía en las organizaciones y empresas de perfeccionar el uso y manejo de la información pues ese interés estuvo determinado no solo por el reconocimiento de esta y el conocimiento, sino también por los beneficios derivados de una adecuada planificación, organización, control de recursos, estrategias y acciones relacionadas con la información, es decir, por una adecuada gestión de este recurso.

La correcta gestión de la información ofrece un entorno de trabajo más organizado, permite tener un mayor control de todas las actividades realizadas en la organización a la información a fin de disminuir el tiempo de realización de muchas tareas.

Su principal beneficio está en permitir el acceso a toda la información que se genera en cada una de las unidades de la organización, el control de versiones, así como la homogeneización de esta información, trayendo consigo un ahorro considerable de los recursos utilizados para el almacenamiento de los archivos y una mejora de los procesos organizacionales.

El surgimiento y avance de las TIC han revolucionado todos los sectores de la sociedad y la economía; y Cuba no está ajena a este desarrollo. Con la creación de la Universidad de las Ciencias Informáticas (UCI) se ha trazado el camino a seguir para lograr una buena gestión de la información. Esta universidad se creó con el objetivo de vincular el estudio-trabajo, poniendo todo el esfuerzo en la creación de software para empresas, tanto cubanas como extranjeras.

En la UCI se han realizado estudios sobre este tema, pero no se cuenta con suficiente experiencia en el campo, aun así, no deja de ser significativo el desarrollo de esta tecnología en la entidad. Por tales motivos se decidió por parte de dicha

universidad y específicamente en la Facultad Regional “Mártires de Artemisa”, desarrollar una aplicación web en tiempo real que gestione la información referente al Ministerio del Comercio Exterior y la Inversión Extranjera (MINCEX).

El MINCEX se crea el 23 de febrero de 1961, y el 2 de marzo de 2009, mediante el Decreto Ley N°. 264, se dispone la creación del nuevo Ministerio del Comercio Exterior y la Inversión Extranjera como Organismo de la Administración Central del Estado, asignándole como objetivo y misión esencial la de preparar y proponer al Gobierno la política integral del Estado y del Gobierno en cuanto a la actividad de comercio exterior, la creación de empresas mixtas, la colaboración económica con otros países, organizaciones y asociaciones extranjeras, y las inversiones que se negocien [1]. Cuenta dentro de su estructura organizativa con 23 direcciones, siendo una de ellas la Dirección de Importaciones.

A esta dirección le corresponde el control de la ejecución de los planes de importaciones, así como es su responsabilidad gestionar todo lo relacionado con los comités de producto, comité de contratación, cartera de proveedores, atención a programas priorizados y la elaboración de dictámenes, manejándose esta información de forma confidencial.

La misma se recibe vía e-mail, telefónica o mediante la entrega personal en formato duro, lo que hace que exista un gran cúmulo de información. Por este motivo la dirección no cuenta con una adecuada forma de gestionarla, necesitándose de mucho tiempo para manejarla. De igual manera, si se desea buscar datos específicos o realizar un reporte determinado se hace de forma manual y, en muchas ocasiones, haciendo consultas en los grandes volúmenes existentes, afectándose la rapidez en la entrega de la información solicitada por lo que existe un bajo nivel en la agilidad en cuanto al desarrollo de los procesos de la dirección. Todo esto es resultado de la gestión manual de la información que hace que el trabajo se dificulte y que los servicios brindados no sean óptimos.

Derivado de la anterior situación problemática se deduce el siguiente **problema de la investigación**: ¿Cómo contribuir a la **agilidad** en el proceso de gestión de la información en la DI del MINCEX?

Se presenta como **objeto de estudio**: proceso de gestión de la información, definiéndose como **campo de acción** el proceso de gestión de la información en el comercio exterior en el área de importaciones.

El **objetivo general** de la investigación es: Desarrollar el servidor para el módulo de la DI para mejorar la **agilidad** en el proceso de gestión de la información en el MINCEX.

**Del anterior objetivo se derivan las siguientes preguntas científicas:**

1. ¿Cuáles son los referentes teóricos que fundamentan los estudios sobre los Sistemas de Gestión de la Información?
2. ¿Cuál es el estado actual que presenta la DI del MINCEX?
3. ¿Cuáles son las metodologías, herramientas y tecnologías a utilizar en el desarrollo de la solución?
4. ¿Cómo implementar el soporte para dar solución a los requerimientos de las aplicaciones clientes para la gestión de información en la DI del MINCEX?
5. ¿Cómo validar el funcionamiento del servidor para el módulo de la DI del MINCEX?

Derivado de estas preguntas científicas se definen las siguientes **tareas de la investigación**:

1. Elaboración del marco teórico y del estado del arte de la investigación.
2. Identificación de los procesos y requerimientos de la DI del MINCEX.
3. Especificación de la metodología, herramientas y tecnologías a utilizar para el desarrollo de la solución propuesta.

4. Realización de los eventos, servicios y plugin en el servidor que den soporte para atender los requerimientos de las aplicaciones clientes para la gestión de información en la DI del MINCEX.

5. Validación del funcionamiento del servidor para módulo de la DI del MINCEX mediante pruebas funcionales al código.

### **Métodos de Investigación empleados: Teóricos y empíricos.**

#### **Métodos teóricos.**

**Analítico-Sintético:** Se empleó para estudiar la influencia de los sistemas de gestión de información sobre las insuficiencias que actualmente se presentan en la dirección del MINCEX, lo cual permitió descubrir y precisar los núcleos fundamentales del problema en la DI. Son métodos complementarios utilizados para la búsqueda y recopilación de datos.

**Análisis Histórico-Lógico:** Este método fue empleado para estudiar el estado del arte y comprender el surgimiento, desarrollo y evolución de los sistemas de gestión de la información.

#### **Métodos Empíricos.**

**Análisis Documental:** Este método es utilizado para obtener información detallada del MINCEX, específicamente de la dirección de importaciones a través de los documentos “Levantamiento de Información” confeccionado previamente por especialistas de la Facultad Regional Mártires de Artemisa de la UCI.

**Aporte práctico:** Consiste en el servidor para el módulo de la DI del sistema de gestión de información del MINCEX.

El presente trabajo de diploma está estructurado por: introducción que se dedica a presentar y fundamentar el problema, así como a exponer los elementos esenciales del diseño teórico y metodológico.

En el **Capítulo 1**: Fundamentación Teórica: se hace un análisis del estado del arte del objeto de estudio, se investiga acerca de los sistemas informáticos vinculados al campo de acción, se fundamentan las metodologías, tecnologías y herramientas utilizadas para el desarrollo del sistema de gestión.

En el **Capítulo 2**: Características, análisis y diseño del sistema: se define el negocio y se describe la solución propuesta para la situación problemática. Se presentan las características y funcionalidades del sistema a partir de los requisitos funcionales y no funcionales capturados. Se realiza el análisis y diseño del sistema.

En el **Capítulo 3**: Implementación y validación del sistema: incluye la programación realizada a partir de los requerimientos y los diagramas del diseño elaborados, así como las métricas y pruebas utilizadas para la validación de la misma.

Finalmente se presentan las conclusiones, referencias bibliográficas, bibliografía y anexos.

# Capítulo 1: Fundamentación teórica, tecnologías, herramientas y metodologías de desarrollo utilizadas.

En el presente capítulo se presentan los principales conceptos asociados al problema y se muestra el estudio a nivel mundial y nacional de varios sistemas existentes. Se expone la metodología de desarrollo de software, las herramientas, tecnologías y el lenguaje de modelado, definiéndose las que serán utilizadas para todo el proceso de desarrollo de la aplicación.

## 1.1 Referentes teóricos

### Información

Para Sergio D' Ambrosio la información “es un conjunto de datos significativos y pertinentes que describen sucesos o entidades.”

El término información, según Verónica Sorberamurina, “alude a un conjunto de datos organizados de manera tal que portan o arrojan un significado.” [2]

Luego de ser analizados los criterios anteriores, la autora está de acuerdo con la definición dada por Verónica Sorberamurina, ya que se puede decir que la información no es más que datos asociados a un contexto determinado los cuales van a suministrar o arrojar algún significado.

### Sistemas de Gestión de Información

Phil Bartle plantea que los sistemas de gestión de la información “son conjuntos de funciones o componentes interrelacionados que forman un todo, obtienen, procesan, almacenan y distribuyen la información, manipulando los datos y consiguiendo, para una organización o empresa, la búsqueda de mejores vías para la dirección y control correspondiente de sus procesos, apoyando la toma de las decisiones en el desempeño de las funciones, de acuerdo con sus propias

estrategias”. [3]

La autora coincide con lo planteado por Phil Bartle ya que los sistemas de gestión de información son la unión de un conjunto de funciones para dar una mejor solución a los problemas de las empresas en este sentido.

### **Módulo**

Según Lastre Reynaldo, un módulo, en programación, “es una parte de un programa de ordenador. De las varias tareas que debe realizar un programa para cumplir con su función u objetivo, un módulo realiza una o varias tareas”.

Este mismo autor considera “que un módulo es un componente auto controlado de un sistema, el cual posee una interfaz bien definida hacia otros componentes; algo es modular si es construido de manera que tal que facilite su ensamblaje, acomodamiento flexible y reparación de sus componentes.” [4]

Lisandra González plantea que “un módulo es una pieza o un conjunto unitario de piezas que, en una construcción, se repiten para hacerla más sencilla, regular y económica. El módulo, por lo tanto, forma parte de un sistema y mantiene algún tipo de relación o vínculo con el resto de los componentes.” [5]

Por lo planteado anteriormente, la autora está de acuerdo con lo afirmado por Lastre Reynaldo, ya que un módulo realiza varias tareas y está diseñado para facilitar la construcción de un sistema en general.

### **Proceso**

“Un proceso es una operación o conjunto combinado de operaciones con datos, o bien una secuencia de acontecimientos definida única y delimitada, que obedece a una intención operacional en condiciones predeterminadas”. [6]

Según el diccionario, “se denomina proceso al conjunto de acciones o actividades sistematizadas que se realizan o tienen lugar con un fin.” [7]

Estas definiciones de proceso se adaptan perfectamente a lo que se busca en esta investigación, ya que un proceso puede definirse como un conjunto de actividades

que se desarrollan para lograr un propósito general.

### **Servidor**

Daniel Rodríguez plantea que, “en informática, un servidor es un tipo de software que realiza ciertas tareas en nombre de los usuarios. El término servidor ahora también se utiliza para referirse al ordenador físico en el cual funciona ese software, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos”. [8]

Existen varios tipos de servidores de acuerdo a las funcionalidades que posee:

- Servidor de aplicaciones.
- Servidor de archivos.
- Servidor de audio/video.
- Servidor de base de datos.
- Servidor de correo.
- Servidor FTP.
- Servidor proxy.
- Servidor web. [9]

La autora coincide con esta definición, ya que, viéndose desde el punto de vista de esta investigación, un servidor se puede definir como un programa que realiza funcionalidades para hacer más fácil el trabajo de los usuarios.

### **Agilidad**

Una definición de agilidad plantea que es la “capacidad de realizar una tarea o acción de la manera más rápida y eficiente”. [10]

Definiéndose en el diccionario como la “facilidad para ejecutar algo de forma rápida, física o mentalmente”. [11]

La autora coincide con lo planteado en ambos conceptos ya que, de forma general, la agilidad no es más que la facilidad de ejecutar un proceso de forma rápida y

eficientemente.

## **1.2 Sistemas de gestión de información para el comercio exterior en el mundo.**

En la actualidad se pueden encontrar en el mundo, diferentes sistemas automatizados que se encargan de gestionar la información relacionada, de una forma u otra, con el comercio exterior, específicamente en el área de las importaciones. Algunos de estos sistemas son:

### **Darsys S.R.L.**

Darsys S.R.L es un software para el comercio exterior desarrollado en Argentina. Está compuesto internamente por varios sistemas de mercado vertical y a medida para agentes de carga internacional, despachantes de aduana, depósitos de zona franca y exportadores e importadores.

El sistema de exportadores e importadores comprende el software de seguimiento de importaciones temporales, soluciones especializadas de exportación e importación, y el software de factura electrónica para exportadores y seguimiento de reintegros. Ofrece una cartera de aplicaciones especializadas que contemplan la rutina diaria de la actividad. Genera actualizaciones periódicas; y, por otra parte, brinda un solvente servicio de soporte. Los sistemas contemplan el envío de información hacia y desde el despachante, con el objeto de agilizar el ingreso de datos y minimizar errores de carga de información. Permite el seguimiento de despachos temporales, con la descarga automática o manual de los insumos a la hora de registrar los permisos de embarque. Además contempla la emisión de informes para el seguimiento de mercaderías y sus tipificaciones, saldos, vencimientos y presentación en aduana.

### **Ventanilla Única del Importador (VUI)**

Desarrollado en Paraguay, VUI es una herramienta de gobierno electrónico que permite a las instituciones intervinientes en el proceso de importación, interactuar

en forma coordinada con la Dirección Nacional de Aduanas, en la gestión de los permisos, autorizaciones y certificaciones, en tiempo real, a través de un Sistema de Gestión Electrónico basado en Internet. Este nuevo sistema permite reducir tiempo y costos de tramitación, mejorar los controles y transparentar la gestión de los procesos que involucran una autorización previa y posterior al despacho de importación. Está dirigido exclusivamente a las importaciones de mercancías que precisen autorizaciones extra aduaneras, simplificando y automatizando los trámites integrados al Sistema SOFIA. Con este proceso, más de 11.000 importadores podrán realizar sus operaciones de manera más segura y transparente, reduciendo trámites y costos.

### **Integration Point Gestión de Importación**

La solución de Gestión de Importación de Integration Point incluye las herramientas necesarias para que un importador administre completamente su proceso de importación, desde la pre-importación hasta la revisión post entrada. Tener una plataforma integrada para administrar todo el proceso de importación permite a las empresas contar con la información regulatoria precisa en el lugar preciso en el momento preciso.

Los beneficios de utilizar Integration Point Gestión de Importación incluyen:

- Mejorar la visibilidad en el proceso de importación completo
- Contar con acceso a información de clasificación global y específica de cada país vía internet
- Utilizar información de comercio internacional actualizada y en tiempo real para la clasificación, preferencias arancelarias, tarifas armonizadas de clasificación, tratados de libre comercio, reglas de origen y más
- Automatizar el procesos (algunas veces manuales y lentos) de validación de la información de entrada
- Dar seguimiento a la comunicación con el agente aduanal

- Integrarse con sistemas ERP para eliminar información duplicada

La solución de Gestión de Importación de Integration Point incluye software para:

- ⤴ Mejorar la visibilidad de entradas así como facilitar el proceso de auditoría post entrada
- ⤴ Clasificar productos correctamente y compartir esta información con todos los socios comerciales
- ⤴ Administrar las calificaciones a los tratados de libre comercio
- ⤴ Ayudar a la recolección y envío de información requerida para el Llenado de Seguridad del Importador (ISF) de los E.U.
- ⤴ Utilizar información de regulaciones comerciales actualizada de más de 140 países

Todos estos sistemas están desarrollados bajo software privativo, por lo que solo pueden servir de estudio para este trabajo, ya que están adaptados a las necesidades del país que los desarrolla. Además no poseen las funcionalidades enfocadas a resolver el problema planteado en esta investigación, por lo tanto no cumple con los requisitos funcionales de la DI, razón por la cual se decide desarrollar el módulo que le dé solución a dichos requisitos

## **1.3 Metodología, herramientas y tecnologías utilizadas para el desarrollo de la solución propuesta.**

### **1.3.1 Metodología de desarrollo.**

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación, para el desarrollo de productos de software. Es donde se va indicando, paso a paso, todas las actividades a realizar para lograr el producto informático deseado, indicando, además, qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener.

## SXP

Es una metodología compuesta por las metodologías SCRUM y XP que ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva, fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo.

Consta de 4 fases principales:

- Planificación-Definición, donde se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto.
- Desarrollo, es donde se realiza la implementación del sistema hasta que esté listo para ser entregado.
- Entrega, puesta en marcha.
- Mantenimiento, donde se realiza el soporte para el cliente.

De cada una de estas fases se realizan numerosas actividades tales como el levantamiento de requisitos, la priorización de la Lista de Reserva del Producto, definición de las Historias de Usuario, Diseño, Implementación, Pruebas, entre otras; de donde se generan artefactos para documentar todo el proceso. Las entregas son frecuentes, y existe una refactorización continua, lo que permite mejorar el diseño cada vez que se le añada una nueva funcionalidad.

SXP está especialmente indicada para proyectos de pequeños equipos de trabajo, rápido cambio de requisitos o requisitos imprecisos, muy cambiantes, donde existe un alto riesgo técnico y se orienta a una estrategia rápida de resultados. Ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro, permitiendo además seguir de forma clara el avance de las tareas a realizar, de forma que se pueda ver día a día cómo progresa el trabajo. **[12]**

### 1.3.2 Lenguaje de Modelado y Herramienta CASE

UML es un lenguaje de modelado para la especificación, visualización, construcción y documentación de los artefactos de un proceso de sistema intensivo. Las herramientas de CASE modelan la información de negocios cuando ésta se transfiere entre distintas entidades organizativas en el seno de una compañía. El objetivo primordial de las herramientas de esta categoría consiste en representar objetos de datos de negocios, sus relaciones, y ayuda a comprender mejor la forma en que fluyen estos objetos de datos entre distintas zonas de negocio.

#### UML 2.0.

UML, por sus siglas en inglés, Unified Modeling Language, es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Está respaldado por el OMG (Object Management Group). 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 y para 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. **[13]**

#### Visual Paradigm 6.4 For UML Enterprise Edition.

Es una herramienta CASE (Ingeniería de Software Asistida por Ordenador) que utiliza UML como lenguaje de modelado. Permite crear diferentes tipos de diagramas en un ambiente totalmente visual. Es muy sencillo de usar, fácil de

instalar y actualizar. Genera código para varios lenguajes.

Visual Paradigm además se integra con herramientas Java como son Eclipse/IBM WebSphere, JBuilder, NetBeans IDE, Oracle JDeveloper, BEA Weblogic. **[14]**

Características de VP 6.4 For UML Enterprise Edition.

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
- Ingeniería inversa - Código a modelo, código a diagrama.
- Ingeniería inversa Java, C++.
- 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.
- 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 (SGBD) existentes a diagramas de Entidad-Relación).
- Generador de informes para generación de documentación.
- Editor de figuras.
- Alta velocidad a la hora de cargar y salvar los proyectos.
- Soporte multilenguaje.

Se escoge para la realización de los diagramas la herramienta Visual Paradigm 6.4,

siguiendo la política establecida en la universidad para el desarrollo de sus productos. Esta herramienta permite de una manera muy fácil desplegar los diagramas necesarios para dar solución al problema, además el equipo de desarrollo presenta conocimientos básicos de la herramienta, lo que posibilita un mejor desempeño con la misma.

### **1.3.3 Entorno de Desarrollo Integrado.**

Un entorno de desarrollo integrado (IDE) es un entorno de programación que ha sido empaquetado como un programa de aplicación, o sea, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

#### **NetBeans.**

Este entorno de desarrollo integrado permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial que lo identifica como módulo. Debido a que estos pueden ser desarrollados de forma independiente, las aplicaciones basadas en esta plataforma pueden ser extendidas fácilmente por otros desarrolladores de software. NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios y una comunidad en constante crecimiento. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio de 2000 y continúa siendo el patrocinador principal de los proyectos. **[15]**

Se elige como entorno de desarrollo NetBeans ya que tiene una interfaz amigable y fácil de comprender aún cuando los usuarios son inexpertos. Posee herramientas para crear aplicaciones profesionales ya sean de escritorio, empresariales o web, no solo en Java sino también en C/C++ y Ruby. Por las características que presenta y por ser además una plataforma flexible, es que se determinó su uso en el desarrollo del servidor para el módulo de la DI del MINCEX.

### 1.3.4 Lenguajes de programación

Los lenguajes de programación son un idioma artificial diseñado para expresar computaciones que pueden ser llevadas a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana.

#### Java

Java es un lenguaje de programación y la primera plataforma informática creada por Sun Microsystems en 1995. Es la tecnología subyacente que permite el uso de programas punteros, como herramientas, juegos y aplicaciones de negocios. Java se ejecuta en más de 850 millones de ordenadores personales de todo el mundo y en miles de millones de dispositivos, como móviles y aparatos de televisión. Existe un gran número de aplicaciones y sitios Web que no funcionan a menos que Java esté instalado, y muchos más que se crean a diario. Java es rápido, seguro y fiable.

#### [16]

Gracias a su versatilidad, eficiencia y portabilidad, Java se ha convertido en un recurso inestimable ya que permite a los desarrolladores [17]:

- Desarrollar software en una plataforma y ejecutarlo en prácticamente cualquier otra plataforma.
- Crear programas para que funcionen en un navegador web y en servicios web.
- Desarrollar aplicaciones para servidores como foros en línea, tiendas, encuestas, procesamiento de formularios HTML, etc.
- Combinar aplicaciones o servicios que usan el lenguaje Java para crear servicios o aplicaciones totalmente personalizados.
- Desarrollar potentes y eficientes aplicaciones para teléfonos móviles,

procesadores remotos, productos de consumo de bajo costo y prácticamente cualquier tipo de dispositivo digital.

Todas las características que posee dan al lenguaje Java las propiedades de robustez y seguridad, evitando, por ejemplo, problemas de desbordamiento de buffer utilizados en ataques a sistemas. Además, la independencia de plataforma hace posible el libre intercambio de software desarrollado en Java sin necesidad de modificaciones.

### **1.3.5 Herramientas de control de versiones**

Cuando se trabaja con archivos, a medida que se van modificando, estos tienen en diferentes momentos, diferentes estados y necesitan ser versionados para mayor seguridad. Entonces se puede decir que el control de versiones consiste en disponer de forma ordenada, de cada una de las versiones o estados de los archivos.

#### **Subversion (SVN)**

Es un sistema de control de versiones diseñado específicamente para reemplazar al popular CVS. Es software libre bajo una licencia de tipo Apache/BSD y se le conoce también como SVN por ser el nombre de la herramienta utilizada en la línea de órdenes.

Subversion puede acceder al repositorio a través de redes, lo que le permite ser usado por personas que se encuentran en distintas computadoras. A cierto nivel, la posibilidad de que varias personas puedan modificar y administrar el mismo conjunto de datos desde sus respectivas ubicaciones, fomenta la colaboración. Se puede progresar más rápidamente sin un único conducto por el cual deban pasar todas las modificaciones. Y puesto que el trabajo se encuentra bajo el control de versiones, no hay razón para temer por que la calidad del mismo vaya a verse afectada. **[18]**

## RapidSVN

Es una interfaz gráfica de usuario para el sistema de revisión de Subversion escrito en C + + utilizando el marco de wxWidgets. Si bien es bastante fácil para los nuevos usuarios de Subversion trabajar con él, también debe ser lo suficientemente potente como para que los usuarios con experiencia sean aún más productivos.

Características:

**Simple:** Proporciona una interfaz fácil de usar para las características de Subversion.

**Eficiente:** Simple para los principiantes pero lo suficientemente flexible como para aumentar la productividad para los usuarios de Subversion con experiencia.

**Portable:** Se ejecuta en cualquier plataforma en la que Subversion y wxWidgets puede ejecutar: Linux, Windows, Mac OS / X, Solaris, etc.

**Rápido:** Completamente escrito en C + +. **[19]**

Se elige Subversion ya que una característica importante es que, a diferencia de CVS, los archivos versionados no tienen cada uno un número de revisión independiente. En cambio, todo el repositorio tiene un único número de versión que identifica un estado común de todos los archivos del repositorio en un instante determinado. Se seleccionó RapidSVN ya que es fácil de usar, tanto por quienes ya conocen Subversion como para quien empieza, pudiendo acceder a direcciones SVN, subir y descargar contenido y sincronizarlo con el servidor original, comprobar su estado, crear y fusionar direcciones, etc. Además tiene la ventaja de funcionar en varias plataformas.

### 1.3.6 Frameworks

Un Framework ofrece componentes como una librería, pero además provee de

plantillas o esqueletos que definen el funcionamiento de las aplicaciones. Estas plantillas que ofrece se pueden adaptar a diferentes necesidades y ahorran trabajo a la hora de escribir una aplicación. Además de que hacen relativamente fácil entender otras aplicaciones hechas con el mismo Framework, ya que comparten un esqueleto similar.

### **JWebSocket**

Es una tecnología orientada al desarrollo de aplicaciones basadas en websockets que gozan de altos niveles de velocidad, escalabilidad y seguridad. jWebSocket permite crear innovadoras aplicaciones HTML5. Los WebSockets reemplazan XHR (Ajax) y Comet por una nueva tecnología de comunicación TCP basada en sockets ultra rápida para la web. La comunicación basada en websockets garantiza menos sobrecarga en la red en el orden de 400 veces y solamente 1/3 de la latencia en la red. Es una solución Open-Source en Java y JavaScript para el protocolo WebSocket de HTML5.

La solución se compone en:

- jWebSocket server: Un servidor desarrollado en Java para llamadas cliente a servidor (C2S), servidor cliente (S2C) y llamadas cliente a cliente (C2C).
- jWebSocket clients: Clientes JavaScript y Java para interacciones con el servidor jWebSocket proporcionando altos niveles de abstracción.
- jWebSocket FlashBridge: Un adaptador basado en flash para navegadores que no son compatibles con HTML5, sin embargo requiere el plug-in the flash instalado en el navegador.

Con jWebSocket se puede crear fácilmente aplicaciones de noticias, monitorización, chat, redes y aplicaciones sociales, juegos en línea o suites de colaboración en línea. Ofrece una amplia gama de funcionalidad desde el

intercambio de bajo nivel basado en Tokens, hasta la sincronización de interfaz gráfica de usuario, las llamadas a procedimiento remoto y mucho más. [20]

Se escoge este framework ya que con jWebSocket se puede programar rápidamente una solución independiente con un mínimo de instalación y configuración o se puede integrar fácilmente en sus aplicaciones existentes. Reutilizando todo el vasto cúmulo de tecnologías, librerías y aplicaciones existentes para la tecnología Java, puede decirse que jWebSocket es una sólida base para la creación de las más diversas aplicaciones web basadas en websockets.

### **1.3.7 Otras herramientas que se utilizan**

#### **Maven**

Maven es una herramienta software para la gestión y construcción de proyectos Java creada por Jason van Zyl, de Sonatype, en 2002. Es similar en funcionalidad a Apache Ant (y en menor medida a PEAR de PHP y CPAN de Perl), pero tiene un modelo de configuración de construcción más simple, basado en un formato XML. Estuvo integrado inicialmente dentro del proyecto Jakarta pero ahora es un proyecto de nivel superior de la Apache Software Foundation.

Maven utiliza un Project Object Model (POM) para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externo, y el orden de construcción de los elementos. Viene con objetivos predefinidos para realizar ciertas tareas claramente definidas, como la compilación del código y su empaquetado. [21]

### **Conclusiones del capítulo**

En el presente capítulo se han dado a conocer los conceptos fundamentales asociados al dominio del problema y se han mostrado varios sistemas existentes y las razones por las cuales no resuelven la situación actual de la DI. También se ha realizado la selección de las metodologías y herramientas a utilizar para el desarrollo de la solución, definiéndose de esta forma como IDE, NetBeans; como

herramienta CASE, VisualParadigm; lenguaje de programación, Java; para el control de versiones, Subversion (SVN) y RapidSVN, y como framework, JwebSocket.

## Capítulo 2. Características, Análisis y Diseño del Sistema.

Este capítulo tiene como objetivo comprender y detallar de forma sencilla el entorno donde está enmarcado el problema. Se conocerá la planificación del proyecto por roles, los requisitos funcionales y no funcionales en los que es necesario basarse para la solución del problema y los diagramas que se generan durante esta fase del proyecto.

### 2.1 Solución Propuesta

Actualmente en la DI, la información se recibe vía e-mail, telefónica o mediante la entrega personal en formato duro, lo que hace que exista un gran cúmulo de esta. Por tal motivo la dirección no cuenta con una adecuada forma de gestionarla, necesiándose de mucho tiempo para manejarla. De igual manera, si se desea buscar datos específicos o realizar un reporte determinado se hace de forma manual y, en muchas ocasiones, haciendo consultas en los grandes volúmenes existentes, afectándose la rapidez en la entrega de la información solicitada, por lo que existe un bajo nivel de agilidad en cuanto al desarrollo de los procesos de la dirección. Todo esto es resultado de la gestión manual de la información que hace que el trabajo se dificulte y que los servicios brindados no sean óptimos

El servidor a implementar permitirá incrementar la agilidad en el proceso de gestión de la información, haciendo que los servicios brindados sean rápidos y se disminuya la pérdida de tiempo que el problema anterior origina.

### 2.2 Información a automatizar

La DI se encarga de seleccionar las empresas proveedoras más viables y contratarlas, así como de aprobar los productos que se importan. Es la encargada, además, de realizar el plan anual de importaciones y controlar de forma segura los

planes enviados por las empresas. Todo esto lo realiza dividido por procesos los cuales llevan las siguientes informaciones:

#### **Comité de contratación**

- Grupo adjunto al comité
- Control de operaciones

#### **Control de proveedores**

- Visto bueno

#### **Planes priorizados**

#### **Plan de importaciones**

**Comité de producto:** En este proceso solo se recibe la información enviada por los clientes y se archiva para después ser utilizada para llenar otro modelo.

## **2.3 Arquitectura y Patrones de Diseño.**

**Arquitectura de Software:** La arquitectura de software de un sistema es la estructura o estructuras del sistema, lo cual abarca componentes de software, las propiedades visibles externamente de esos componentes, y las relaciones entre ellas.

**Patrón de diseño:** Es una solución a un problema de diseño no elemental que es segura y reusable. Los patrones de diseño, facilitan el aprendizaje al programador inexperto, pudiendo establecer parejas problema-solución. En la programación orientada a objetos resulta complicado descomponer el sistema en objetos, los patrones de diseño permitirán identificar a los objetos apropiados de una manera mucho más sencilla. Además, los patrones de diseño, también ayudarán a definir las interfaces, identificando los elementos clave de las mismas y las relaciones existentes entre ellas. De igual modo nos facilitará la especificación de las

implementaciones. De forma casi automática, nos ayudan a reutilizar código, facilitando la decisión entre herencia o composición y proporcionando reusabilidad, extensibilidad y mantenimiento.

### **Arquitectura en n-capas**

Lo que se conoce como arquitectura en capas es en realidad un estilo de programación donde el objetivo principal es separar los diferentes aspectos del desarrollo, tales como las cuestiones de presentación, lógica de negocio, mecanismos de almacenamiento, etc. Se debe resaltar el uso del término "capa" y no el de "nivel" porque no significan lo mismo. El término capa se utiliza para referenciar a las distintas partes en que una aplicación se divide desde un punto de vista lógico; mientras que nivel corresponde a la forma física en que se organiza una aplicación.

Capa de presentación: es la que se encarga de que el sistema interactúe con el usuario y viceversa, muestra el sistema al usuario, le presenta la información y obtiene la información del usuario en un mínimo de proceso. En el mundo de la informática es conocida como interfaz gráfica y debe tener la característica de ser amigable, o sea, entendible y fácil de usar para el usuario. Esta capa se comunica únicamente con la capa intermedia o de negocio.

Capa de negocio: es donde residen las funciones que se ejecutan, se reciben las peticiones del usuario, se procesa la información y se envían las respuestas tras el proceso. Se denomina capa de negocio o capa de lógica del negocio, porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de acceso a datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él. [22]

Capa de acceso a datos: Es una porción de código que justamente realiza el acceso a los datos. De esta manera cuando es necesario cambiar el motor de base

de datos, solamente se corregirá esa capa.

Capa de datos: Es donde están los datos y se corresponde directamente con la definición de esquemas, tablas, vistas, procedimientos almacenados y todo lo que se pueda o deba poner en un motor de base de datos. [23]

## 2.4 Planificación del Proyecto por roles.

Tabla 1: Planificación del proyecto por roles

Rol	Responsabilidad	Nombre
<b>Gerente (Manager)</b>	Se encarga de dirigir y controlar las tareas del equipo. Es el responsable de tomar las decisiones finales, acerca de estándares y convenciones a seguir durante el proyecto.	Yeilin Martínez Torres
<b>Líder del Proyecto (Scrum Master)</b>	Es el encargado coordinar y facilitar las reuniones, de asegurar que se consigan los objetivos de la reunión planificados para la iteración. Determina cuándo es necesario realizar algún cambio para lograr los objetivos de cada iteración.	Annie Cedeño López
<b>Cliente (Customer)</b>	Participa en las tareas que	

	involucran la lista de reserva del producto, aprueba las modificaciones de esta y las de la iteración.	
<b>Programadores (Programmers)</b>	Es el encargado de producir el código y escribir las pruebas unitarias. Debe existir una comunicación y coordinación adecuada entre los programadores y otros miembros del equipo.	Yisel Meléndez Puentes Yelina Rodríguez Pérez Cristina I. Rangel Falber
<b>Analista (Analyst)</b>	Es el encargado de escribir las historias de usuario y las pruebas funcionales para validar su implementación. Asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio, todo esto lo realiza junto con el cliente.	Yelina Rodríguez Pérez Cristina I. Rangel Falber
<b>Diseñadores (Designers)</b>	Es el encargado del diseño del sistema; así como el de los prototipos de interfaces, máximos responsables de la realización del diseño de las metáforas y supervisan el	Yelina Rodríguez Pérez Cristina I. Rangel Falber

	proceso de construcción.	
<b>Encargado de Pruebas (Tester)</b>	Es el encargado de ayudar al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.	Yelina Rodríguez Pérez Cristina I. Rangel Falber
<b>Arquitecto (Architect)</b>	Su trabajo tiene que ver con la estructura y el diseño del sistema, por eso el arquitecto se vincula directamente con el analista y el diseñador. Ayuda en el diseño de las metáforas.	Yelina Rodríguez Pérez Cristina I. Rangel Falber
<b>Diseñador de Base de Datos</b>	Es el encargado de diseñar la base de datos.	Yisel Meléndez Puentes

## 2.5 Modelo de Dominio.

La metodología SXP define diversas actividades para realizar una detallada descripción del negocio. Una de ellas es el Modelo de Dominio, el cual se realiza cuando dicho negocio no está bien definido entre los clientes y los desarrolladores del proyecto. A continuación se presente el diagrama de dominio para el sistema que se propone donde, si una empresa extranjera desea importar bienes y servicios hacia Cuba, lo realiza mediante la dirección de importaciones que se encuentra en el MINCEX. Esta empresa debe llenar el perfil de control de proveedores, una vez

llo este, pasa a los grupos de comité de contratación que se realizan en cada una de las direcciones del MINCEX. Estos grupos analizan este perfil y realizan un acta donde emiten criterios de evolución de estas empresas, dichas actas son enviadas al comité de contratación que es la dirección de importaciones. Esta dirección analiza estas actas y plasman el criterio general en un acta dando la aprobación o no, si es aceptada se le envía el Visto Bueno a la empresa. Uno de los planes que se analizan en la dirección son los priorizados y los de importaciones que son enviados por las empresas hacia ella, realizando con ellos un plan total por cada uno.

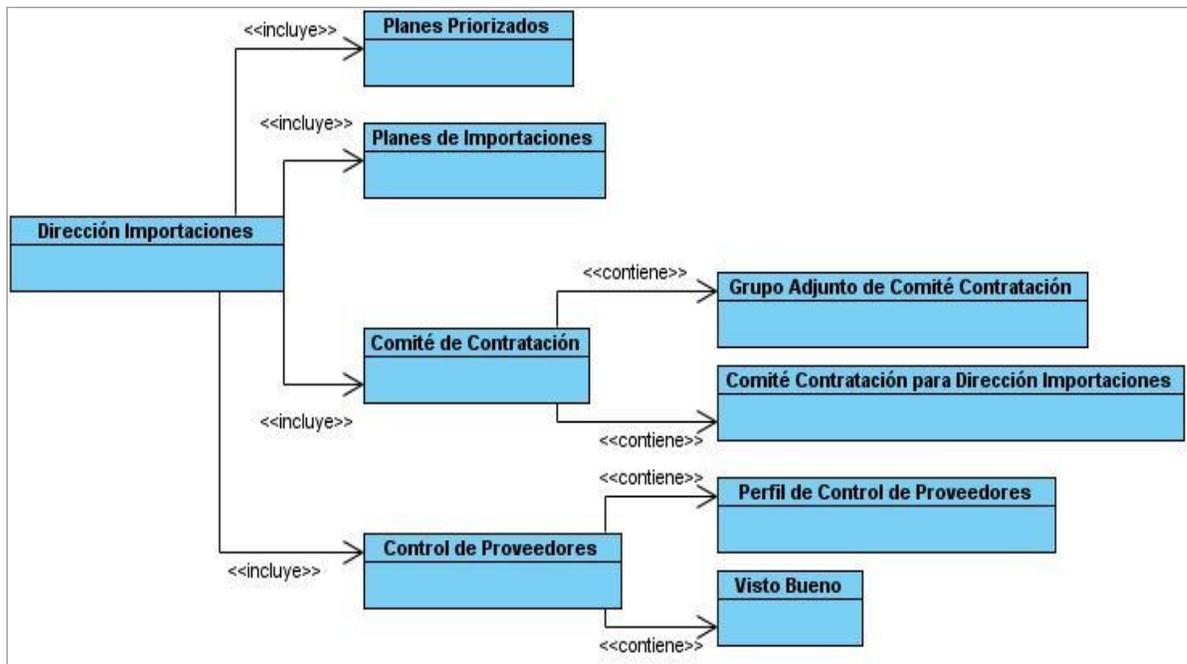


Figura 1: Modelo de dominio

## 2.6 Lista de Reserva del Producto (LRP).

Otra de las actividades más importantes que define la metodología SXP es la Lista de Reserva del Producto (LRP), en la cual se recoge en una lista priorizada todo el trabajo a desarrollar en el proyecto. Esta lista puede crecer y modificarse a medida

que se obtienen más conocimientos acerca del producto y del cliente. El objetivo es asegurar que el producto definido al terminar la lista sea el más óptimo posible y para esto la lista debe acompañar los cambios en el entorno y el producto.

Tabla 2: Lista de reserva del producto

Prioridad	Ítem	Descripción	Estimación	Estimado por
<b>Muy Alta</b>				
	1	Insertar control del plan de importaciones.	2 días	ANALISTA
	2	Modificar control del plan de importaciones.	2 días	ANALISTA
	3	Eliminar control del plan de importaciones.	1 días	ANALISTA
	4	Insertar control de planes priorizados.	2 días	ANALISTA
	5	Modificar control de planes priorizados.	2 días	ANALISTA
	6	Eliminar control de planes priorizados.	1 días	ANALISTA
	7	Archivar comité de productos.	4 días	ANALISTA
	8	Insertar control de proveedores del sistema MINCEX.	2 días	ANALISTA
	9	Modificar control de proveedores del sistema MINCEX.	2 días	ANALISTA
	10	Eliminar control de proveedores del sistema MINCEX.	1 días	ANALISTA
	11	Insertar comité de contratación.	2 días	ANALISTA
	12	Modificar comité de contratación.	2 días	ANALISTA

	13	Eliminar comité de contratación.	1 días	ANALISTA
<b>Alta</b>				
	14	Buscar control del plan de importaciones.	2 días	ANALISTA
	15	Buscar control de planes priorizados.	2 días	ANALISTA
	16	Buscar control de proveedores del sistema MINCEX.	2 días	ANALISTA
	17	Buscar comité de contratación.	2 días	ANALISTA
<b>Media</b>				
	18	Realizar reporte del control del plan de importaciones.	2 días	ANALISTA
	19	Mostrar el control del plan de importaciones.	2 días	ANALISTA
	20	Realizar reporte del control de planes priorizados.	2 días	ANALISTA
	21	Mostrar control de planes priorizados.	2 días	ANALISTA
	22	Realizar reporte del control de proveedores del sistema MINCEX.	2 días	ANALISTA
	23	Mostrar control de proveedores del sistema MINCEX.	2 días	ANALISTA
	24	Realizar reporte del comité de contratación.	2 días	ANALISTA
	25	Mostrar comité de contratación.	2 días	ANALISTA
<b>Baja</b>				

26	Imprimir control del plan de importaciones.	1 días	ANALISTA
27	Exportar control del plan de importaciones a formato pdf.	1 días	ANALISTA
28	Exportar control del plan de importaciones a formato excel.	1 días	ANALISTA
29	Imprimir control de planes prioritizados.	1 días	ANALISTA
30	Exportar control de planes prioritizados a formato pdf.	1 días	ANALISTA
31	Exportar control de planes prioritizados a formato excel.	1 días	ANALISTA
32	Imprimir control de proveedores del sistema MINCEX.	1 días	ANALISTA
33	Exportar control de proveedores del sistema MINCEX a formato pdf.	1 días	ANALISTA
34	Exportar control de proveedores del sistema MINCEX a formato excel.	1 días	ANALISTA
35	Imprimir comité de contratación.	1 días	ANALISTA
36	Exportar comité de contratación a formato pdf.	1 días	ANALISTA
37	Exportar comité de contratación a formato excel.	1 días	ANALISTA
<b>RNF (Requisitos No Funcionales)</b>			
38	Sistema Operativo Microsoft Windows Millennium, XP, Seven o		

		Linux.		
	39	Postgres 8.4+ permite el uso de Bases de datos relacionales.		
	40	Mozilla Firefox 6.0+ o Internet Explorer 6.0+.		
	41	Procesador: Pentium IV 1.8 GHz, Disco Duro: 80 Gb, PC con 512 MB(o superior) de memoria RAM para el desarrollo de la aplicación.		
	42	El sistema garantizará la confidencialidad, integridad y disponibilidad de la información almacenada en la base de datos.		
	43	La información estará protegida contra accesos no autorizados utilizando mecanismos de autenticación y autorización que puedan garantizar el cumplimiento de esto: usuario, contraseña y nivel de acceso dado por la asignación de roles.		
	44	La aplicación podrá ser utilizada por los usuarios una vez que se le haya impartido un entrenamiento sobre la misma, contará con una ayuda con el fin de documentar al usuario de su utilización.		
	45	El sistema debe permitir a los		

		usuarios conectarse desde cualquier computadora que esté conectada a la red dentro del ministerio.	
	46	El sistema no debe tener más de 1 hora de no disponibilidad en el mes.	
	47	La aplicación debe tener una interfaz sencilla, agradable, legible y de fácil uso para el usuario. El contenido será mostrado de manera comprensible y fácil de leer.	
	48	Los colores a emplear en el sistema son el azul y el blanco que están relacionados con la entidad.	
	49	La interfaz principal contará con el logotipo de la entidad.	

## 2.7 Historias de Usuarios y Tareas de Ingeniería.

Las historias de usuarios en la metodología de desarrollo SXP, son las que describen las tareas que el sistema debe hacer, cuestión que depende en gran medida de las especificaciones realizadas por el cliente. Se escriben con un lenguaje natural y con palabras concisas para no exceder su tamaño en unas pocas líneas de texto. Van a ser la guía para la construcción posterior de las pruebas de aceptación comprobando de esta manera la correcta implementación de las historias de usuario. A continuación se muestran las historias de usuario que se

generan durante el proceso del **Control de proveedores**. Para más información ver Anexo 6: Historias de Usuario.

Tabla 3: Historias de usuario y Tareas de la ingeniería

Historia de Usuario	
<b>Número:</b> HU_1	<b>Nombre Historia de Usuario:</b> Gestionar control de proveedores del sistema MINCEX.
<b>Modificación de Historia de Usuario Número:</b> 1	
<b>Usuario:</b> Cristina I. Rangel Falber Yelina Rodríguez Pérez Yisel Meléndez Puentes	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 5/5
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 5/5
<b>Descripción:</b> La presente historia de usuario tiene como objetivos insertar, modificar y eliminar los perfiles de las empresas importadoras y los vistos buenos de aprobación de los nuevos proveedores.	
<b>Observaciones:</b> Para que esto sea posible es necesario estar autenticado en el sistema, además para modificar y eliminar deben estar archivados los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores. Esta HU hace referencia a los requisitos 8, 9 y 10.	
<b>Prototipo de interfase:</b> Ver Anexo 1: Prototipo de Interfaz Gestionar control de proveedores del sistema MINCEX.	
Tarea de Ingeniería	
<b>Número Tarea:</b> 1.1	<b>Número Historia de Usuario:</b> HU_1
<b>Nombre Tarea:</b> Implementación de la funcionalidad insertar control de proveedores del sistema MINCEX.	

<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 2/5
<b>Fecha Inicio:</b> 11/01/2012	<b>Fecha Fin:</b> 12/01/2012
<b>Programador Responsable:</b> Yelina Rodríguez Pérez. Cristina I. Rangel Falber.	
<b>Descripción:</b> Implementar las funcionalidades necesarias que permitan al usuario insertar la nueva información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	

Tarea de Ingeniería	
<b>Número Tarea:</b> 1.2	<b>Número Historia de Usuario:</b> HU_1
<b>Nombre Tarea:</b> Implementación de la funcionalidad modificar control de proveedores del sistema MINCEX.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 2/5
<b>Fecha Inicio:</b> 13/01/2012	<b>Fecha Fin:</b> 16/01/2012
<b>Programador Responsable:</b> Yelina Rodríguez Pérez. Cristina I. Rangel Falber.	
<b>Descripción:</b> Implementar las funcionalidades necesarias que permitan al usuario modificar la información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	

Tarea de Ingeniería	
<b>Número Tarea:</b> 1.3	<b>Número Historia de Usuario:</b> HU_1
<b>Nombre Tarea:</b> Implementación de la funcionalidad eliminar control de proveedores del sistema MINCEX.	

<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 1/5
<b>Fecha Inicio:</b> 17/01/2012	<b>Fecha Fin:</b> 17/01/2012
<b>Programador Responsable:</b> Yelina Rodríguez Pérez. Cristina I. Rangel Falber.	
<b>Descripción:</b> Implementar las funcionalidades necesarias que permitan al usuario eliminar la información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	

<b>Historia de Usuario</b>	
<b>Número:</b> HU_2	<b>Nombre Historia de Usuario:</b> Buscar control de proveedores del sistema MINCEX.
<b>Modificación de Historia de Usuario Número:</b> 1	
<b>Usuario:</b> Cristina I. Rangel Falber Yelina Rodríguez Pérez Yisel Meléndez Puentes	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 2/5
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 2/5
<b>Descripción:</b> La presente historia de usuario tiene como objetivo buscar los perfiles de las empresas importadoras y los vistos buenos de aprobación de los nuevos proveedores.	
<b>Observaciones:</b> Para que esto sea posible es necesario estar autenticado en el sistema y además tener archivados los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores. Esta HU hace referencia al requisito 17.	
<b>Prototipo de interfase:</b> Ver Anexo 2: Prototipo de Interfaz Buscar control de proveedores del sistema	

MINCEX.

Tarea de Ingeniería	
<b>Número Tarea:</b> 2.1	<b>Número Historia de Usuario:</b> HU_2
<b>Nombre Tarea:</b> Implementación de la funcionalidad buscar control de proveedores del sistema MINCEX.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 2/5
<b>Fecha Inicio:</b> 8/02/2012	<b>Fecha Fin:</b> 9/02/2012
<b>Programador Responsable:</b> Yelina Rodríguez Pérez. Cristina I. Rangel Falber.	
<b>Descripción:</b> Implementar las funcionalidades necesarias que permitan al usuario buscar la información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	

Historia de Usuario	
<b>Número:</b> HU_3	<b>Nombre Historia de Usuario:</b> Realizar reporte del control de proveedores del sistema MINCEX.
<b>Modificación de Historia de Usuario Número:</b> 1	
<b>Usuario:</b> Cristina I. Rangel Falber Yelina Rodríguez Pérez Yisel Meléndez Puentes	<b>Iteración Asignada:</b> 3
<b>Prioridad en Negocio:</b> Media	<b>Puntos Estimados:</b> 2/5
<b>Riesgo en Desarrollo:</b> Medio	<b>Puntos Reales:</b> 2/5
<b>Descripción:</b> La presente historia de usuario tiene como objetivo realizar reportes de los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	

**Observaciones:** Para que esto sea posible es necesario estar autenticado en el sistema y además tener archivados los perfiles de las empresas importadoras y los esos los cuales llevan las siguientes informaciones vistos buenos de aprobación de los nuevos proveedores. Esta HU hace referencia al requisito 23.

**Prototipo de interfase:**

Ver Anexo 3: Prototipo de Interfaz Realizar reporte del control de proveedores del sistema MINCEX y Mostrar control de proveedores del sistema MINCEX.

<b>Tarea de Ingeniería</b>	
<b>Número Tarea:</b> 3.1	<b>Número Historia de Usuario:</b> HU_3
<b>Nombre Tarea:</b> Implementación de la funcionalidad realizar reporte del control de proveedores del sistema MINCEX.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 2/5
<b>Fecha Inicio:</b> 20/02/2012	<b>Fecha Fin:</b> 21/02/2012
<b>Programador Responsable:</b> Yelina Rodríguez Pérez. Cristina I. Rangel Falber.	
<b>Descripción:</b> Implementar las funcionalidades necesarias que permitan al usuario realizar reporte de la información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	

<b>Historia de Usuario</b>	
<b>Número:</b> HU_4	<b>Nombre Historia de Usuario:</b> Mostrar control de proveedores del sistema MINCEX.
<b>Modificación de Historia de Usuario Número:</b> 1	
<b>Usuario:</b> Cristina I. Rangel Falber Yelina Rodríguez Pérez	<b>Iteración Asignada:</b> 3

Yisel Meléndez Puentes	
<b>Prioridad en Negocio:</b> Media	<b>Puntos Estimados:</b> 1/5
<b>Riesgo en Desarrollo:</b> Medio	<b>Puntos Reales:</b> 1/5
<b>Descripción:</b> La presente historia de usuario tiene como objetivo mostrar los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	
<b>Observaciones:</b> Para que esto sea posible es necesario estar autenticado en el sistema y además tener archivados los perfiles de las empresas importadoras y los vistos buenos de aprobación de los nuevos proveedores. Esta HU hace referencia al requisito 24.	
<b>Prototipo de interfase:</b> Ver Anexo 3: Prototipo de Interfaz Realizar reporte del control de proveedores del sistema MINCEX.	

Tarea de Ingeniería	
<b>Número Tarea:</b> 4.1	<b>Número Historia de Usuario:</b> HU_4
<b>Nombre Tarea:</b> Implementación de la funcionalidad mostrar control de proveedores del sistema MINCEX.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 1/5
<b>Fecha Inicio:</b> 22/02/2012	<b>Fecha Fin:</b> 22/02/2012
<b>Programador Responsable:</b> Yelina Rodríguez Pérez. Cristina I. Rangel Falber.	
<b>Descripción:</b> Implementar las funcionalidades necesarias que permitan al usuario mostrar la información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	

Historia de Usuario	
<b>Número:</b> HU_5	<b>Nombre Historia de Usuario:</b> Imprimir control de proveedores del sistema MINCEX.
<b>Modificación de Historia de Usuario Número:</b> 1	
<b>Usuario:</b> Cristina I. Rangel Falber Yelina Rodríguez Pérez Yisel Meléndez Puentes	<b>Iteración Asignada:</b> 4
<b>Prioridad en Negocio:</b> Baja	<b>Puntos Estimados:</b> 1/5
<b>Riesgo en Desarrollo:</b> Bajo	<b>Puntos Reales:</b> 1/5
<b>Descripción:</b> La presente historia de usuario tiene como objetivo imprimir los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.	
<b>Observaciones:</b> Para que esto sea posible es necesario estar autenticado en el sistema y además tener archivados los perfiles de las empresas importadoras y los vistos buenos de aprobación de los nuevos proveedores. Esta HU hace referencia al requisito 37.	
<b>Prototipo de interfase:</b> Ver Anexo 4: Prototipo de Interfaz Imprimir control de proveedores del sistema MINCEX.	

Tarea de Ingeniería	
<b>Número Tarea:</b> 5.1	<b>Número Historia de Usuario:</b> HU_5
<b>Nombre Tarea:</b> Implementación de la funcionalidad imprimir control de proveedores del sistema MINCEX.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 1/5
<b>Fecha Inicio:</b> 7/03/2012	<b>Fecha Fin:</b> 7/03/2012

**Programador Responsable:** Yelina Rodríguez Pérez.  
Cristina I. Rangel Falber.

**Descripción:** Implementar las funcionalidades necesarias que permitan al usuario imprimir la información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores.

<b>Historia de Usuario</b>	
<b>Número:</b> HU_6	<b>Nombre Historia de Usuario:</b> Exportar control de proveedores del sistema MINCEX.
<b>Modificación de Historia de Usuario Número:</b> 1	
<b>Usuario:</b> Cristina I. Rangel Falber Yelina Rodríguez Pérez Yisel Meléndez Puentes	<b>Iteración Asignada:</b> 4
<b>Prioridad en Negocio:</b> Baja	<b>Puntos Estimados:</b> 2/5
<b>Riesgo en Desarrollo:</b> Bajo	<b>Puntos Reales:</b> 2/5
<b>Descripción:</b> La presente historia de usuario tiene como objetivo exportar los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores a formato PDF y Excel.	
<b>Observaciones:</b> Para que esto sea posible es necesario estar autenticado en el sistema y además tener archivados los perfiles de las empresas importadoras y los vistos buenos de aprobación de los nuevos proveedores. Esta HU hace referencia a los requisitos 38 y 39.	
<b>Prototipo de interfase:</b> Ver Anexo 5: Prototipo de Interfaz Exportar control de proveedores del sistema MINCEX.	

Tarea de Ingeniería	
<b>Número Tarea:</b> 6.1	<b>Número Historia de Usuario:</b> HU_6
<b>Nombre Tarea:</b> Implementación de la funcionalidad exportar control de proveedores del sistema MINCEX.	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 2/5
<b>Fecha Inicio:</b> 8/03/2012	<b>Fecha Fin:</b> 9/03/2012
<b>Programador Responsable:</b> Yelina Rodríguez Pérez. Cristina I. Rangel Falber.	
<b>Descripción:</b> Implementar las funcionalidades necesarias que permitan al usuario exportar la información referente a los perfiles de las empresas importadoras y los vistos buenos de aprobación de los proveedores a formato PDF y Excel.	

## 2.8 Plan de Release.

En este paso se define el plan de release para realizar las entregas intermedias y la entrega final. El mismo tiene como entrada la relación de Historias de Usuario definidas previamente. Para colocar una historia en cada iteración se tiene en cuenta la prioridad definida por el cliente. Como resultado de la priorización de historias se llegó a la siguiente planificación.

Tabla 4: Plan de release

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
2da Iteración	En esta iteración se desarrollarán las historias de usuario que tienen prioridad alta.	1,2,7,8,13,14,1 9,20, 25	6 semanas y 2 días.
3era Iteración	En esta iteración se desarrollarán las historias de usuario de prioridad media y se irá integrando a las ya realizadas.	3,4,9,10,15,16, 21,22	3 semanas y 1 día.
4ta Iteración	En esta iteración se desarrollarán las historias de usuario de prioridad media y se integrarán a las historias de usuario ya implementadas para conformar el módulo para la Dirección de Importaciones del MINCEX.	5,6,11,12,17,1 8,23, 24	2 semanas y 2 días.

## 2.9 Diseño con Metáfora.

El diseño con metáforas es, sencillamente, el diseño de la solución más simple que pueda funcionar y ser implementada en un momento dado del proyecto; lo cual genera el artefacto conocido como Modelo de Diseño, que está compuesto por un diagrama de paquetes que representa esta solución. A continuación se representa el diagrama de paquetes para el sistema que se propone, que se usa para reflejar la organización de los paquetes y sus elementos.

Los paquetes View, Controller, Model y Service representan las 4 capas de la arquitectura del proyecto. El paquete nombrado ExtJS4 representa a las librerías

del Framework de Javascript ExtJs. El paquete View contiene los paquetes necesarios donde se incluyen las vistas, modelos y controladores del sistema. Dentro del sub-paquete view se encuentran todas las vistas del sistema. El sub-paquete controller contiene los controladores mediante el cual se medarán a ejecutar las diferentes funcionalidades del sistema. Dentro del sub-paquete model estarán los modelos creados.

El paquete Server Side incluye los sub-paquetes controller, service y model del lado del servidor. El sub-paquete controller contiene los plugins y eventos que se encargarán de ejecutar las funcionalidades del sistema haciendo uso del sub-paquete service el cual brindará los servicios necesarios para satisfacer los requisitos del sistema. Dentro del sub-paquete model se ubicarán todas las entidades y acceso a datos de la aplicación. El paquete Libs contiene todas las librerías del lado del servidor necesarias para la construcción de la aplicación. El paquete jWebSocket Config contiene los xml de configuración del sistema los cuales permitirán vincular las vistas de la aplicación con el servidor.

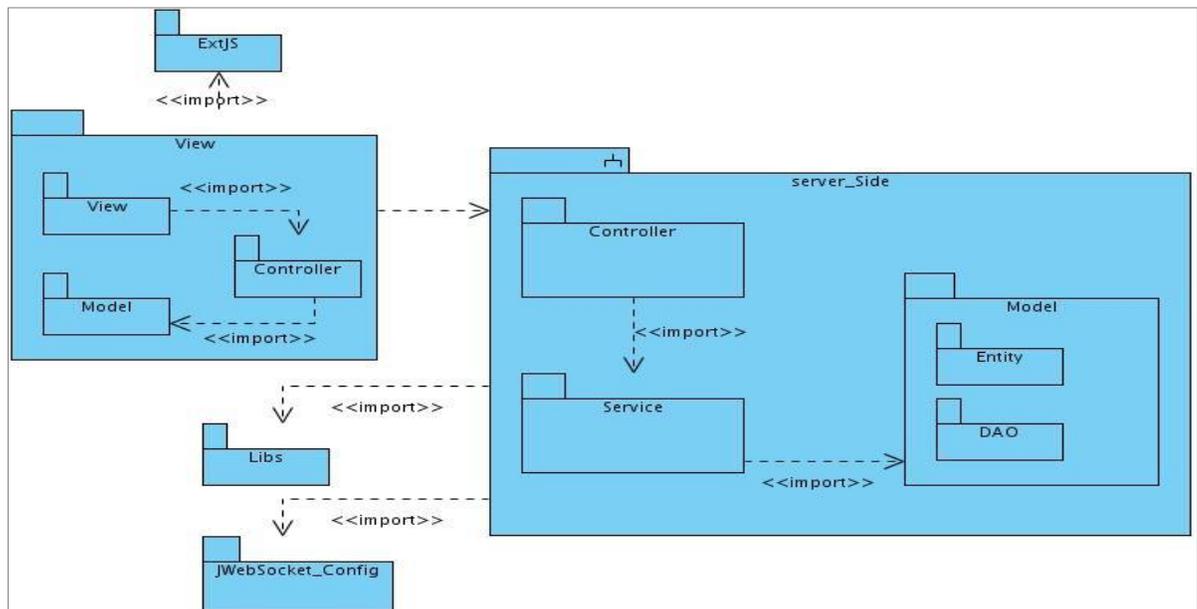


Figura 2: Diseño con metáfora

## 2.10 Diagrama de Componentes.

En el diagrama de componentes se describen los elementos físicos del sistema y sus relaciones. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, entre otros. En el diagrama de componentes se muestran un conjunto de paquetes, componentes y las relaciones entre estos, que están vinculadas al sistema. El paquete nombrado ExtJS4 representa a las librerías del Framework de Javascript ExtJs. El paquete View contiene los paquetes necesarios donde se incluyen las vistas, modelos y controladores del sistema. Dentro del sub-paquete view se encuentran todas las vistas del sistema. El sub-paquete controller contendrá los controladores mediante el cual se mandarán a ejecutar las diferentes funcionalidades del sistema. Dentro del sub-paquete model estarán los modelos creados.

El servidor se encuentra ubicado específicamente en el paquete Server\_Side. El mismo, internamente, está compuesto por el sub-paquete Controller que es donde se encuentran los plugin y los eventos creados y que sirven de intermediarios entre las funcionalidades implementadas y la aplicación cliente. Este sub-paquete, se relaciona con el paquete Service, que está compuesto por los servicios, que son los que implementan las funcionalidades propias del sistema. Este último tiene una relación con el sub-paquete Model que posee en su interior las entidades y el archivo que permite a la funcionalidad acceder a la base de datos. De forma general, el paquete Server\_Side se relaciona con los paquetes Libs y jWebSocket\_config. En el primero se ubican las librerías a utilizar, particularmente las de jWebSocket e Hibernate y en el segundo se ubican los archivos de configuración del framework jWebSocket que harán que la aplicación funcione correctamente. A continuación se presenta el diagrama de componentes para el sistema que se propone:



## **Conclusiones del capítulo.**

En el presente capítulo se definieron las historias de usuario y las tareas de la ingeniería asociadas a ellas, las que permitieron delimitar las actividades a desarrollar para lograr el producto deseado. Se identificaron los requisitos funcionales y no funcionales los cuales pueden estar sujetos a cambios durante el desarrollo de la solución, así como los diagramas necesarios para comenzar la siguiente fase.

## Capítulo 3 Implementación y validación del sistema

En el presente capítulo se brinda una explicación de la implementación realizada del lado del servidor y la validación del mismo mediante las pruebas de caja blanca, que ayudaran a detectar errores en el código. Se dan a conocer, además, los resultados y funcionalidades obtenidas, así como los aportes sociales y económicos de la investigación.

### 3.1 Estándar de código

- Los nombres de las variables, métodos, carpetas empezarán con minúsculas y si estos identificadores están compuestos por varias palabras las siguientes empezaran con mayúscula.
- Los nombres de las clases, stores, models empezarán con mayúscula y si estos identificadores están compuestos por varias palabras las siguientes empezaran con mayúscula.
- Los comentarios de implementación se deberán utilizar al comienzo de cada archivo y antes de cada método, los caracteres de estos comentarios que se utilizan son `/**/` para comentarios en bloque y `//` para comentarios de una sola línea.

### 3.2 Implementación del lado del servidor

El servidor es el encargado de responder las peticiones enviadas por los clientes. Para ello define eventos, plugins y servicios que obtienen esta información para ser procesada en el mismo.

#### 3.2.1 Eventos de la aplicación

Los eventos de la aplicación heredan de la clase `C2SEvent` importado de la librería `org.jwebsocket.eventmodel.event.C2SEvent` de `jwebsocket`. Los eventos `C2SEvent/S2CEvent` son definiciones orientadas a objetos usado en el `EventModel`

para el tratamiento o manejo de las variables de entrada y de salida desde el cliente o para el cliente. Otra observación es el uso de anotaciones como la que se encuentra encima de las properties de los eventos `@ImportFromToken` mediante la cual se logra la asignación automática de campos de los eventos, además de copiar o mover el valor de los tokens de entrada a los campos delimitados en los eventos. Esta anotación debe estar definida para cada uno de los atributos del evento, bien se puede declarar encima de la propiedad set o encima de la declaración de los propios atributos.

Se definieron un total de 33 eventos para desarrollar la solución, mostrándose a continuación un ejemplo de su implementación:

### Evento `addApproval`

En este caso el evento o la operación que se desea realizar es la de añadir un visto bueno, por lo que se necesita definir los parámetros de entrada que el cliente le envía al servidor para que luego sean almacenados en la base de datos.

```

public class addApproval extends C2SEvent{
    String date;
    String requestingCompany;
    String supplier;
    String country;
    String product;
    String companyClassif;
    String clasificacion;
    String observations;
    String name;
    String lastName;
    String state;

    public String getLastName() {
        return lastName;
    }

    @ImportFromToken
    public void setLastName(String lastName) {
        this.lastName = lastName;
    }

    /*Aquí estan ocultas las properties que faltan por mostrar */
    public String getName() {...}
}

```

Figura 4: Evento `addApproval`

### 3.2.2 Clases services de la aplicación.

Las clases services permiten realizar la operación solicitada por el cliente. Estas clases son las que ordenan la ejecución de consultas en la bases de datos mediante el uso del dao genérico que tiene como atributo. Se implementaron 7 servicios, mostrándose a continuación un ejemplo de ellos:

#### Servicio ApprovalService.

En esta clase se implementan los requisitos o funcionalidades involucradas con los vistos buenos. Es el encargado de enviarle la respuesta esperada por el cliente en cada llamado realizado.

Métodos implementados en la clase ApprovalService:

Tabla 5: Métodos implementados en la clase ApprovalService

<p><b>public String Add(addApproval approval)</b></p> <p>Este método recibe como parámetro el evento pertinente con los datos entrados por el cliente. Verifica que no exista la empresa que se desea entrar y retorna una cadena con el mensaje correspondiente.</p>
<p><b>public String Update(modifyApproval approval)</b></p> <p>Este método recibe como parámetro el evento pertinente con los datos entrados por el cliente. Verifica que la empresa a modificar se encuentre y retorna una cadena con el mensaje correspondiente.</p>
<p><b>public List&lt;DimportApproval&gt; Find(searchApproval approval)</b></p> <p>Este método recibe como parámetro el evento pertinente con los datos entrados por el cliente. Se busca según el criterio deseado, retornándose una lista con el resultado.</p>
<p><b>public String Delete(deleteApproval approval)</b></p> <p>Este método recibe como parámetro el evento pertinente con los datos entrados por el cliente. Se elimina según el identificador de los datos, retornándose el mensaje pertinente.</p>

```
public List<DimportApproval> ShowAll(showApproval approval)
```

Este método recibe como parámetro un evento vacío. Se buscan todos los datos de que se encuentran en la base de datos, retornándose una lista con el resultado.

```
public void setDao(GenericDao dao)
```

Esta property permite asignarle la clase GenericDao al servicio para poder ejecutar las consultas con hibernate en la base de datos.

### 3.2.3 Clases plugin de la aplicación.

Los plugins son extensiones para dar soporte personalizados (C2S/S2C) a los acontecimientos en el lado del servidor, básicamente son los listeners para escuchar o notificar a los eventos relacionados en el mismo.

Cada clase plugin hereda de la clase EventModelPlugin del paquete de jWebSocket org.jwebsocket.eventmodel.plugin.EventModelPlugin, además de implementar los distintos processEvent que se encargarán de escuchar las notificaciones realizadas por el cliente. Fueron definidos un total de 7 plugins, mostrándose un ejemplo a continuación.

#### Plugin ApprovalPlugin.

Este plugin es el encargado de manejar los distintos eventos involucrados con los vistos buenos. Permite añadir, actualizar, buscar, eliminar y mostrar los mismos. Esta clase hace uso del servicio ApprovalService definido con anterioridad.

Métodos implementados en la clase ApprovalPlugin:

Tabla 6: Métodos implementados en la clase ApprovalPlugin

```
public void processEvent(addApproval aEvent, C2SResponseEvent aResponse)
```

Invoca al método Add del servicio ApprovalService para insertar los datos del visto bueno. Mediante el aResponse envía una cadena como respuesta al cliente notificando si la operación realizada fue satisfactoria o no.

**public void processEvent(modifyApproval aEvent, C2SResponseEvent aResponse)**

Invoca al método Update del servicio ApprovalService para modificar los datos del visto bueno. Mediante el aResponse envía una cadena como respuesta al cliente notificando la operación realizada.

**public void processEvent(searchApproval aEvent, C2SResponseEvent aResponse)**

Invoca al método Find del servicio ApprovalService para buscar los datos solicitados. Mediante el aResponse envía una lista con los datos encontrados como respuesta al cliente.

**public void processEvent(deleteApproval aEvent, C2SResponseEvent aResponse)**

Invoca al método Delete del servicio ApprovalService para eliminar los datos del visto bueno. Mediante el aResponse envía una cadena como respuesta al cliente notificando la operación realizada

**public void processEvent(showApproval aEvent, C2SResponseEvent aResponse)**

Invoca al método ShowAll del servicio ApprovalService para mostrar todos los datos del visto bueno. Mediante el aResponse envía una lista con los datos como respuesta al cliente.

**public void setService(ApprovalService service)**

Se utiliza para inyectarle al plugin el servicio que utilizará la clase, en este caso requiere de un ApprovalService.

```

public class ApprovalPlugin extends EventModelPlugIn {
    private ApprovalService service;

    public void processEvent(addApproval aEvent, C2SResponseEvent aResponse) {
        aResponse.getArgs().setBoolean("success", true);
        aResponse.getArgs().setString("infomation", service.Add(aEvent));
    }

    public void processEvent(modifyApproval aEvent, C2SResponseEvent aResponse) {
        aResponse.getArgs().setString("infomation", service.Update(aEvent));
        aResponse.getArgs().setBoolean("success", true);
    }

    public void processEvent(searchApproval aEvent, C2SResponseEvent aResponse) {
        aResponse.getArgs().setList("newList", service.Find(aEvent));
        aResponse.getArgs().setBoolean("success", true);
    }

    public void processEvent(deleteApproval aEvent, C2SResponseEvent aResponse) {...}

    public void processEvent(showApproval aEvent, C2SResponseEvent aResponse) {...}

    public void setService(ApprovalService service) {
        this.service = service;
    }
}

```

Figura 5: Plugin ApprovalPlugin

### 3.3 Pruebas al código

La etapa de pruebas es una de las fases del ciclo de vida de los proyectos. Se le podría ubicar después del análisis, el diseño y la programación, pero dependiendo del proyecto en cuestión y del modelo de proceso elegido, su realización podría ser en forma paralela a las fases citadas o inclusive repetirse varias veces durante la duración del proyecto. La importancia de esta fase será mayor o menor según las características del sistema desarrollado, llegando a ser vital en sistemas de tiempo

real u otros en los que los errores sean irre recuperables.

Las pruebas no tienen el objeto de prevenir errores sino de detectarlos. Se efectúan sobre el trabajo realizado y se deben encarar con la intención de descubrir la mayor cantidad de errores posible. Aunque no hay una clasificación oficial o formal acerca de los diversos tipos de pruebas de software, existen dos vertientes fundamentales:

- Pruebas de tipo Caja Negra (Black Box testing): cuando una aplicación es probada usando su interfaz externa, generalmente la GUI.
- Pruebas de tipo Caja Blanca (White Box testing): cuando una aplicación es probada desde dentro, usando su lógica aplicativa. [24]

### 3.3.1 Pruebas de caja Blanca

Para probar el código se han realizado las pruebas de caja blanca debido a que es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivarlos. Este tipo de prueba define un conjunto de métodos a utilizar:

- Prueba del camino básico
- Prueba de condición
- Prueba del flujo de datos
- Prueba de bucles

Es importante destacar que las pruebas de caja blanca intentan garantizar que:

- Se ejecutan al menos una vez todos los caminos independientes de cada módulo.
- Se utilizan las decisiones en su parte verdadera y en su parte falsa.
- Se ejecuten todos los bucles en sus límites.
- Se utilizan todas las estructuras de datos internas.

Se escoge para probar el código, de las pruebas de caja blanca, la técnica del camino básico ya que permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. Los casos de prueba obtenidos de este conjunto garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa.

Para hacer uso de esta técnica se debe dividir el código a probar en nodos para después representarlos en un grafo de flujo, que se utiliza para trazar más fácilmente los caminos definidos. Posteriormente, se calcula la complejidad ciclomática, que representa el número de caminos independientes del conjunto básico de un programa. Esta medida ofrece al probador de software un límite superior para el número de pruebas que debe realizar para garantizar que se ejecutan por lo menos una vez cada sentencia. Si la cantidad de caminos independientes son menores que la complejidad, el código probado no es eficiente, por lo que la funcionalidad tampoco lo es y se debe volver a implementar. Se realizaron un total de 33 pruebas haciendo uso de esta técnica, mostrándose a continuación un ejemplo de las mismas:

```

public String Add(addApproval approval) {
1  List<DimportApproval> listtt =
   dao.findByParamAndValue(DimportApproval.class, "requestingCompany",
   approval.getRequestingCompany());
2  if (listtt.isEmpty()) {
3
   DimportApproval approvalToSave = new DimportApproval();
   approvalToSave.setDate(approval.getDate());

   approvalToSave.setRequestingCompany(approval.getRequestingCompany());
   approvalToSave.setSupplier(approval.getSupplier());
   approvalToSave.setCountry(approval.getCountry());
   approvalToSave.setProduct(approval.getProduct());
   approvalToSave.setCompanyClassif(approval.getCompanyClassif());
   approvalToSave.setClasification(approval.getClasification());
   approvalToSave.setObservations(approval.getObservations());
   approvalToSave.setName(approval.getName());
   approvalToSave.setLastName(approval.getLastName());
   approvalToSave.setState(approval.getState());
   dao.save(approvalToSave);
   return "Se ha insertado satisfactoriamente";
4  return "La empresa ya existe";
   }
5  }

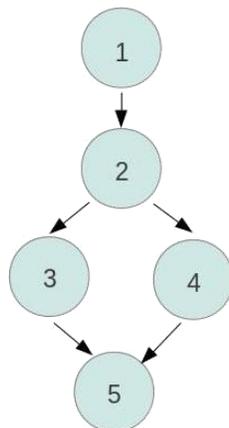
```

Complejidad Ciclomática= 2

Caminos Básicos

C1: 1-2-3-5

C2: 1-2-4-5



Caso de Prueba para C1	
listtt.isEmpty()	true
ApprovalToSave approvalToSave.setDate() approvalToSave.setRequestingCompany() approvalToSave.setSupplier() approvalToSave.setCountry() approvalToSave.setProduct() approvalToSave.setCompanyClassif() approvalToSave.setClasification() approvalToSave.setObservations(); approvalToSave.setName(); approvalToSave.setLastName(); approvalToSave.setState();	new DimportApproval() approval.getDate() approval.getRequestingCompany() approval.getSupplier() approval.getCountry() approval.getProduct() approval.getCompanyClassif() approval.getClasification() approval.getObservations() approval.getName() approval.getLastName() approval.getState()
return	"Se ha insertado satisfactoriamente"
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que los datos se han insertado satisfactoriamente.

Caso de Prueba para C2	
listtt.isEmpty()	false
return	"La empresa ? ya existe"
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que la empresa que se ha entrado ya existe.

```

public String Update(modifyApproval approval) {
    Long idApproval = Long.parseLong(approval.getIdApproval());
    DimportApproval approvalToUpdate =
    (DimportApproval)dao.findByPK(DimportApproval.class, idApproval);
1  List<DimportApproval> listtt =
    dao.findByParamAndValue(DimportApproval.class, "requestingCompany",
    approval.getRequestingCompany());
2  if (listtt.isEmpty()) {
3      approvalToUpdate.setDate(approval.getDate());

    approvalToUpdate.setRequestingCompany(approval.getRequestingCompany());
    approvalToUpdate.setSupplier(approval.getSupplier());
    approvalToUpdate.setCountry(approval.getCountry());
    approvalToUpdate.setProduct(approval.getProduct());
    approvalToUpdate.setCompanyClassif(approval.getCompanyClassif());
    approvalToUpdate.setClasification(approval.getClasification());
    approvalToUpdate.setObservations(approval.getObservations());
    approvalToUpdate.setName(approval.getName());
    approvalToUpdate.setLastName(approval.getLastName());
    approvalToUpdate.setState(approval.getState());

    dao.update(approvalToUpdate);
    return "Se ha actualizado satisfactoriamente";
4  return "La empresa ya existe";
5  }
}

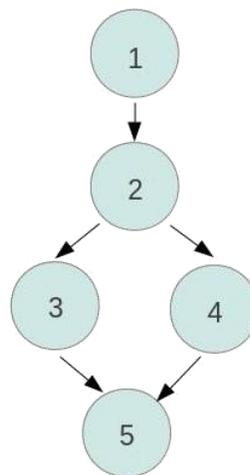
```

Complejidad Ciclomática= 2

Caminos Básicos

C1: 1-2-3-5

C2: 1-2-4-5



Caso de Prueba para C1	
idApproval	approval.getIdApproval()
approvalToUpdate	dao.findByPK(DimportApproval.class, idApproval)
listtt.isEmpty()	true
approvalToUpdate.setDate() approvalToUpdate.setRequestingCompany() approvalToUpdate.setSupplier() approvalToUpdate.setCountry() approvalToUpdate.setProduct() approvalToUpdate.setCompanyClassif() approvalToUpdate.setClasification() approvalToUpdate.setObservations() approvalToUpdate.setName() approvalToUpdate.setLastName() approvalToUpdate.setState()  return	approval.getDate() approval.getRequestingCompany() approval.getSupplier() approval.getCountry() approval.getProduct()  approval.getCompanyClassif() approval.getClasification() approval.getObservations() approval.getName() approval.getLastName() approval.getState()  "Se ha actualizado satisfactoriamente"
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que los datos se han insertado satisfactoriamente.

Caso de Prueba para C2	
idApproval	approval.getIdApproval()
approvalToUpdate	dao.findByPK(DimportApproval.class, idApproval)
listtt.isEmpty()	false
return	"La empresa solicitante ? ya existe"
Resultado	El método ha recorrido el camino adecuado devolviendo un mensaje que la empresa que se desea actualizar ya existe.

### 3.3.2 Resultados de las pruebas

Se realizaron 33 pruebas en total, de ellas el 100% arrojaron resultados satisfactorios con el objetivo de encontrar la mayor cantidad de errores existentes y darle solución a los mismos. A continuación se muestra una tabla para ver como fue el comportamiento de dichas pruebas.

Tabla 7: Resultados obtenidos en las pruebas

Cantidad de pruebas	Tipo de prueba	Resultados obtenidos
33	Caja blanca	100% satisfactorios

### 3.4 Resultados obtenidos

Como resultado de este trabajo, el servidor para el módulo de la DI del MINCEX queda disponible en su versión 1.0. Se obtuvo un servidor que cumple con las especificaciones del cliente desde el punto de vista de las funcionalidades. Se logró un alto grado de optimización utilizándose los componentes propios del framework jWebSocket logrando una reutilización del código a gran escala.

### 3.5 Funcionalidades obtenidas

Entre las principales funcionalidades que posee el Módulo para la DI del MINCEX en su versión 1.0 se pueden mencionar:

- Es capaz de insertar, actualizar, eliminar y buscar la información referente a cada uno de los procesos definidos para esta dirección.
- Permite mostrar la información perteneciente a dichos procesos.
- Posibilita la creación de reportes necesarios para la DI.
- Brinda la posibilidad de mostrar, de exportar a pdf o a excel e imprimir la información de estos procesos.
- Archiva los documentos excel enviados por las empresas en el sistema.

### **3.6 Aporte social y económico**

Con la utilización del módulo de la DI, se logrará eliminar la poca seguridad, el atraso de las entregas de informaciones y la pérdida de tiempo que son unos de los mayores problemas que existe en dicha dirección, trayendo consigo una mejor utilización de la información. Con la aplicación, el personal de la DI podrá enfocarse principalmente en satisfacer las necesidades que presentan los clientes con los que realizan los contratos. Se debe decir que esta aplicación se creó con el objetivo de contribuir a mejorar la calidad del trabajo del personal de la DI, tratando de lograr un mejor uso de la información con la que se trabaja en la dirección y la satisfacción de los mismos. La creación del servidor ayudó con la informatización del MINCEX, logrando con esto que se realicen menos gastos de material de oficina, permitiendo un ahorro económico para el ministerio

### **Conclusiones del capítulo**

Durante el desarrollo del capítulo se presentaron fragmentos de código utilizados en la implementación del servidor y se ejemplificaron las pruebas realizadas al mismo. Se expuso, como resultado, el servidor para el módulo de la DI dándose a conocer sus principales funcionalidades y, finalmente, se brinda el aporte social y económico de la investigación.

## Conclusiones generales

En la presente investigación de manera general, se logró dar cumplimiento a al objetivo general y a las tareas de la investigación, arribándose a las conclusiones siguientes

- Se realizó un estudio de las herramientas a nivel nacional e internacional, evidenciándose la necesidad de crear una solución que cumpla con los requerimientos funcionales de la DI.
- Se seleccionaron las herramientas, tecnologías y metodologías que brindaron el soporte necesario para lograr un producto de calidad y rendimiento acordes a las necesidades de la Dirección.
- Se desarrolló el servidor que dio solución a los requerimientos planteados por del cliente.
- Se validó el eficaz funcionamiento del servidor mediante pruebas funcionales al código.

## Referencias bibliográficas

1. “¿Quiénes Somos?”, n.d.,  
<http://www.mincex.cu/index.php/quienessomos.html>.
2. Romero Heber, R. Y. 2009. “Gestión del conocimiento y la información en el Polo de Gestión Universitaria”. Ciudad de la Habana, Universidad de las Ciencias Informáticas.
3. Bartle, Phil. 2009. “INFORMACIÓN PARA LA GESTIÓN y gestión de la información”. Potenciación comunitaria.  
<http://www.scn.org/mpfc/modules/mon-miss.htm>. Citado: Sábado, 17/12/2011
4. Lastre Reynaldo, Reyvis.E.2008. “Análisis y diseño del Módulo de Reportes para el portal de la UJC en la UCI”. Universidad de las Ciencias en Informática de la Habana, 2008. Citado: Domingo 01/01/ 2012.
5. González Serrano, Lisandra.2010. “Implementación del Subsistema de Generación de Informes Selux“. Ing. Bernardo Zaragoza Hijuelos, Raudi. A Bacallao Sánchez. Universidad de las Ciencias en Informática de la Ciudad Habana, 2010. Citado: Sábado, 17/12/2011.
6. “Definición de Proceso - Diccionario de Términos técnicos de Internet - Glosario“. <http://tecnologia.glosario.net/terminos-tecnicos-internet/proceso-1360.html>.Citado: Viernes 18 /05/2012
7. DefiniciónABC. “Definición de Proceso “. <http://www.definicionabc.com/general/proceso.php>. Citado: Viernes 01 /06/2012
8. Rodríguez, Daniel Omar .2008. “HARDENING APACHE“. <http://danielomarrodriguez.blogspot.com/2008/01/hardening-apache.html>  
Citado: Viernes, 08 /06/ 2012.
9. ¿Cuántos tipos de Servidores existen?.  
[http://www.soporteavanzado.com/jm/index.php?option=com\\_content&view=](http://www.soporteavanzado.com/jm/index.php?option=com_content&view=)

- article&id=10%3Acuantos-tipos-de-servidores-existe&catid=32%3Alanguages&Itemid=41&lang=es. Citado: Viernes, 08 /06/ 2012
10. "Tus preguntas".  
<http://tuspreguntas.misrespuestas.com/preg.php?idPregunta=9628>. Citado: Viernes 08 /06/2012.
  11. "Diccionarios".[http://diccionarios.elmundo.es/diccionarios/cgi/diccionario/lee\\_diccionario.html?busca=agilidad&diccionario=1&submit=Buscar+](http://diccionarios.elmundo.es/diccionarios/cgi/diccionario/lee_diccionario.html?busca=agilidad&diccionario=1&submit=Buscar+). Citado: Viernes 08 /06/2012.
  12. Leyva Samada, Lisandra Isabel. 2009. *Flujo de investigación para la metodología ágil SXP*. Tesis de ingeniería inédita. Universidad de las Ciencias Informáticas. Cuba, 2009.
  13. Rojas González, Arletis. "UML" - EcuRed.  
<http://www.ecured.cu/index.php/UML>. Citado: Lunes, 16 /1/2012.
  14. Hernadis, J.A. 2005. "Versión Cero: Visual Paradigm for UML".  
<http://www.versioncero.com/noticia/210/visual-paradigm-for-uml>. Citado: Domingo, 15 /1/2012.
  15. González López, Lusy Leydis. "NetBeans" - EcuRed.  
<http://www.ecured.cu/index.php/NetBeans>. Citado: Lunes 16 /1/2012.
  16. "¿Qué es Java y por qué lo necesito?".  
[http://www.java.com/es/download/faq/whatis\\_java.xml](http://www.java.com/es/download/faq/whatis_java.xml). Citado: Domingo, 15 /1/2012.
  17. "Conozca más sobre la tecnología Java". <http://java.com/es/about/>. Citado: Domingo 15 /1/2012.
  18. "Subversion - Guía Ubuntu" . 2009. <http://www.guia-ubuntu.org/index.php?title=Subversion>. Citado: Domingo 15 /1/2012.
  19. "RapidSVN". 2011. <http://www.rapidsvn.org/>. Citado: Domingo 15 /1/2012.
  20. Websocket Community. ¿Qué es jWebSocket? » jWebSocket: "A dream is

- coming true....” . 2011. <http://softwarelibre.hab.uci.cu/jwsblog/?p=1>. Citado: Domingo 15 /1/2012.
21. "Maven 1.x - Welcome". <http://maven.apache.org/maven-1.x/>. Citado: Domingo 15 /1/2012.
22. Hernández Suárez, José de Jesús .2008. «Enter@te». <http://www.enterate.unam.mx/Articulos/2006/febrero/arquitect.htm>. Citado: Sábado, 17/12/2011
23. Tentor, Julio. “Software y Aplicaciones Web”. <http://www.jtentor.com.ar/post/Arquitectura-de-N-Capas-y-N-Niveles.aspx> . Citado Martes, 22/5/2012.
24. Katz-Lichtenstein .2003. “Black Box Web Testing with HttpUnit“. <http://onjava.com/onjava/2003/05/07/blackboxwebtest.html>. Citado: Martes 22 /5/2012.

## Bibliografía

1. “Comercio Exterior”, n.d., [http://www.cubagob.cu/rel\\_ext/mincex/com\\_ext.htm](http://www.cubagob.cu/rel_ext/mincex/com_ext.htm). Citado: Domingo 18/12/11.
2. “Evolución de los Sistemas de Información”, n.d., <http://eradelsaber.bligoo.com/content/view/302880/Evolucion-de-los-Sistemas-de-Informacion.html>. Citado: Domingo 18/12/11.
3. “Importancia de la gestión de la información y el conocimiento en el proceso de cambio organizacional - Monografias.com”, n.d., <http://www.monografias.com/trabajos70/gestion-informacion-proceso-cambio-organizacional/gestion-informacion-proceso-cambio-organizacional.shtml>. Citado: Domingo 18/12/11.
4. “La información a través del tiempo - Evolución de la información - Wikilearning”, n.d., [http://www.wikilearning.com/articulo/la\\_informacion\\_a\\_traves\\_del\\_tiempo/8411](http://www.wikilearning.com/articulo/la_informacion_a_traves_del_tiempo/8411). Citado: Domingo 18/12/11.
5. “La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual- Archivo de artículos del Observatorio para la CiberSociedad”, n.d., <http://www.cibersociedad.net/archivo/articulo.php?art=218>. Citado: Domingo 18/12/11.
6. “¿Qué son los sistemas de gestión?”, n.d., <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>. Citado: Domingo 18/12/11.
7. “Diseño del sistema de gestión de información del Centro de Estudios de Medio Ambiente y Recursos Naturales (CEMARNA) de la Universidad de Pinar Del Río | GestioPolis“. <http://www.gestiopolis.com/administracion-estrategia/sistemas-de-gestion-de-informacion-en-estudio-de-medio-ambiente.htm>. Citado: Domingo

- 18/12/11.
8. Gonzalez Almora, Aymelis. Abigantus Perez, Pedro.2010. *Implementación de servicios para la gestión de información en el polo Petrosoft*. Tesis de ingeniería inédita. Universidad de las Ciencias Informáticas. Cuba, 2010.
  9. Kelly Naranjo, Julio Antonio. Cañabate Gutierrez, Julio Cesar. 2010. *Sistema para la Gestión de la Información de los Juegos Deportivos Inter-Facultades Versión 2.0*. Tesis de ingeniería inédita. Universidad de las Ciencias Informáticas. Cuba, 2010.
  10. Ortiz Valmaseda, Marcos. Rodriguez Cervantes, Susany. 2009. *Herramienta para la migración de bases de datos Oracle a PostgreSQL*. Tesis de ingeniería inédita. Universidad de las Ciencias Informáticas. Cuba, 2009.
  11. Pedroso Marquez, Dismey. Ulloa Gomez, Ernesto Wilfredo. 2011. *Sistema de Gestión de Información para las Coordinaciones Regionales de Prevención del Delito de la República Bolivariana de Venezuela. Módulos: Atención de Casos, Bienes Muebles y Materiales*. Tesis de ingeniería inédita. Universidad de las Ciencias Informáticas. Cuba, 2011.
  12. "Software de gestión para importadores y exportadores". <http://www.darsys.com.ar/software-importadores-exportadores.html> Citado: Miércoles 6/06/2012
  13. "Ventanilla única del importador (VUI) ". <http://www.aduana.gov.py/125-6-ventanilla-unica-del-importador--vui-.html>. Citado: Miércoles 6/06/2012
  14. "IntegrationPoint". <http://www.integrationpoint.com/es/solutions/importmanagement.html>. Citado: Miércoles 6/06/2012
  15. Geronimo Manso. "Control de versiones para archivos y documentos – Herramientas". <http://www.geronet.com.ar/?p=415>. Citado: Lunes 16 /1/2012.
  16. GNUSTEP. "¿Qué es un Framework?. <http://gnustep.wordpress.com/gnustep-a-fondo/%C2%BFque-es-un-framework-%C2%BFcomo-se-utiliza/>. Citado: Lunes 16 /1/2012.
  17. Alezenni Sablón. "Lenguaje de Programación – EcuRed ".

- [http://www.ecured.cu/index.php/Lenguaje\\_de\\_Programaci%C3%B3n](http://www.ecured.cu/index.php/Lenguaje_de_Programaci%C3%B3n). Citado: Lunes 16 /1/2012
18. Yuliet Legra. “IDE de Programación – EcuRed “. [http://www.ecured.cu/index.php/IDE\\_de\\_Programaci%C3%B3n](http://www.ecured.cu/index.php/IDE_de_Programaci%C3%B3n). Citado: Lunes 16 /1/2012
19. Rafael Barzanallana. “Apuntes Informática Aplicada a la Gestión Pública. Capítulo 2, Ingeniería del software, metodologías de desarrollo. 2011/12. Universidad de Murcia (España).Rafael Barzanallana“. <http://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-de-desarrollo.html>. Citado: viernes 18 /11/2011
20. “Herramientas CASE. Ingeniería del software. Informática Aplicada a la gestión Pública. Universidad de Murcia“. [http://www.um.es/docencia/barzana/IAGP/Enlaces/CASE\\_principales.html](http://www.um.es/docencia/barzana/IAGP/Enlaces/CASE_principales.html). Citado: viernes 18 /11/2011
21. Métodos de prueba de caja blanca. <http://gemini.udistrital.edu.co/comunidad/grupos/arquisoft/fileadmin/Estudiantes/Pruebas/HTML%20-%20Pruebas%20de%20software/node26.html>. Citado: Jueves, 17/05/2012.
22. Polo Usaola, Macario. “Pruebas de Sistemas de Información”. Departamento de tecnologías y sistemas de información. Universidad de Castilla- La Mancha.
23. “¿Quiénes Somos?”, n.d., <http://www.mincex.cu/index.php/quienessomos.html>.
24. Romero Heber, R. Y. 2009. “Gestión del conocimiento y la información en el Polo de Gestión Universitaria“. Ciudad de la Habana, Universidad de las Ciencias Informáticas.
25. Bartle, Phil. 2009. “INFORMACIÓN PARA LA GESTIÓN y gestión de la información”. Potenciación comunitaria. <http://www.scn.org/mpfc/modules/monmiss.htm>. Citado: Sábado, 17/12/2011
26. Lastre Reynaldo, Reyvis.E.2008. “Análisis y diseño del Módulo de Reportes para el portal de la UJC en la UCI“. Universidad de las Ciencias en Informática de la Habana, 2008. Citado: Domingo 01/01/ 2012.

27. González Serrano, Lisandra. 2010. "Implementación del Subsistema de Generación de Informes Selux". Ing. Bernardo Zaragoza Hijuelos, Raudi. A Bacallao Sánchez. Universidad de las Ciencias en Informática de la Ciudad Habana, 2010. Citado: Sábado, 17/12/2011.
28. "Definición de Proceso - Diccionario de Términos técnicos de Internet - Glosario". <http://tecnologia.glosario.net/terminos-tecnicos-internet/proceso-1360.html>. Citado: Viernes 18 /05/2012
29. DefiniciónABC. "Definición de Proceso". <http://www.definicionabc.com/general/proceso.php>. Citado: Viernes 01 /06/2012
30. Rodríguez, Daniel Omar .2008. "HARDENING APACHE". <http://danielomarrodriguez.blogspot.com/2008/01/hardening-apache.html> Citado: Viernes, 08 /06/ 2012.
31. ¿Cuántos tipos de Servidores existen?. [http://www.soporteavanzado.com/jm/index.php?option=com\\_content&view=article&id=10%3Acuantos-tipos-de-servidores-existe&catid=32%3Alenguajes&Itemid=41&lang=es](http://www.soporteavanzado.com/jm/index.php?option=com_content&view=article&id=10%3Acuantos-tipos-de-servidores-existe&catid=32%3Alenguajes&Itemid=41&lang=es). Citado: Viernes, 08 /06/ 2012
32. "Tus preguntas". <http://tuspreguntas.misrespuestas.com/preg.php?idPregunta=9628>. Citado: Viernes 08 /06/2012.
33. "Diccionarios". [http://diccionarios.elmundo.es/diccionarios/cgi/diccionario/lee\\_diccionario.html?busca=agilidad&diccionario=1&submit=Buscar+](http://diccionarios.elmundo.es/diccionarios/cgi/diccionario/lee_diccionario.html?busca=agilidad&diccionario=1&submit=Buscar+). Citado: Viernes 08 /06/2012.
34. Leyva Samada, Lisandra Isabel. 2009. *Flujo de investigación para la metodología ágil SXP*. Tesis de ingeniería inédita. Universidad de las Ciencias Informáticas. Cuba, 2009.
35. Rojas González, Arletis. "UML" - EcuRed. <http://www.ecured.cu/index.php/UML>. Citado: Lunes, 16 /1/2012.
36. Hernadis, J.A. 2005. "Versión Cero: Visual Paradigm for UML". <http://www.versioncero.com/noticia/210/visual-paradigm-for-uml>. Citado: Domingo,

- 15 /1/2012.
37. González López, Lusy Leydis. "NetBeans" - EcuRed.  
<http://www.ecured.cu/index.php/NetBeans>. Citado: Lunes 16 /1/2012.
38. "¿Qué es Java y por qué lo necesito?".  
[http://www.java.com/es/download/faq/whatis\\_java.xml](http://www.java.com/es/download/faq/whatis_java.xml). Citado: Domingo, 15 /1/2012.
39. "Conozca más sobre la tecnología Java". <http://java.com/es/about/>. Citado: Domingo 15 /1/2012.
40. "Subversion - Guía Ubuntu" . 2009. <http://www.guia-ubuntu.org/index.php?title=Subversion>. Citado: Domingo 15 /1/2012.
41. "RapidSVN". 2011. <http://www.rapidsvn.org/>. Citado: Domingo 15 /1/2012.
42. Websocket Community. ¿Qué es jWebSocket? » jWebSocket: "A dream is coming true...." . 2011. <http://softwarelibre.hab.uci.cu/jwsblog/?p=1>. Citado: Domingo 15 /1/2012.
43. "Maven 1.x - Welcome" . <http://maven.apache.org/maven-1.x/>. Citado: Domingo 15 /1/2012.
44. Hernández Suárez, José de Jesús .2008. «Enter@te». <http://www.enterate.unam.mx/Articulos/2006/febrero/arquitect.htm>. Citado: Sábado, 17/12/2011
45. Tentor, Julio. "Software y Aplicaciones Web". <http://www.jtentor.com.ar/post/Arquitectura-de-N-Capas-y-N-Niveles.aspx> . Citado Martes, 22/5/2012.
46. Katz-Lichtenstein .2003. "Black Box Web Testing with HttpUnit". <http://onjava.com/onjava/2003/05/07/blackboxwebtest.html>. Citado: Martes 22 /5/2012.