



Universidad del Azuay
Facultad de Ciencias de la Administración
Escuela de Ingeniería de Sistemas

**Desarrollo de una aplicación de Facturación Electrónica
utilizando herramientas de Gestión e Integración
en Entornos Java.**

**Trabajo de graduación previo a la obtención del título de
Ingeniero en Sistemas**

Autores:

Diego Alfredo Enderica Ortega

Diego Fernando Pazhe Hugo

Director:

Ing. Andrés Patiño

Cuenca, Ecuador

2017

DEDICATORIA

Dedico este trabajo principalmente a Dios, quien me ha permitido tener familia, amigos y seres queridos los cuales siempre me han impulsado a seguir adelante, a mis padres quienes me han inculcado grandes valores, ofreciéndome su apoyo incondicional para lograr alcanzar cada meta propuesta, a mis hermanos que siempre han estado apoyándome, creyendo en mí, brindándome toda su colaboración y confianza para poder culminar esta meta.

Diego Enderica Ortega

DEDICATORIA

A Dios por permitirme llegar a culminar esta etapa tan importante en mi vida, a mis padres: Carlos y Fanny que me apoyaron durante toda una vida, siendo el apoyo más fuerte e importante, son mis principales pilares, a mis hermanos: Gabriel, Roberto, Freddy y Norma que de una u otra forma siempre estuvieron junto a mí durante este bonito proceso, a mi tío Jaime que a la distancia siempre estuvo aconsejándome, a los demás familiares y amigos que a lo largo de mi carrera estuvieron dando su apoyo y animándome a seguir adelante.

Diego Pazhe Hugo

AGRADECIMIENTO

Agradecemos a la Universidad del Azuay, a la Escuela de Ingeniería de Sistemas, a todos los docentes por la valiosa instrucción académica brindada, de manera especial al Ing. Andrés Patiño por todo el apoyo y colaboración brindada durante el desarrollo de la presente tesis.

ÍNDICE DE CONTENIDOS

DEDICATORIA.....	ii
AGRADECIMIENTO	iv
ÍNDICE DE CONTENIDOS.....	V
ÍNDICE DE ILUSTRACIONES	VIII
ÍNDICE DE TABLAS.....	XI
RESUMEN.....	XIII
ABSTRACT	XIV
CAPÍTULO 1	1
TECNOLOGÍAS APLICADAS	1
1.1 INTRODUCCIÓN	1
1.2 MAVEN	1
1.3 HIBERNATE	2
1.3.1 CARACTERÍSTICAS HIBERNATE:	3
1.3.2 COMPARACIÓN ENTRE EL API DE PERSISTENCIAS DE JAVA (JPA) E HIBERNATE	4
1.4 JAVA SERVER FACES JSF	4
1.4.1 ¿QUE ES JAVA SERVER FACES?	5
1.4.2 CARACTERÍSTICAS DE JAVA SERVER FACES.....	5
1.4.3 VERSIONES.....	6
1.5 PRIME FACES.....	6
1.5.1 CARACTERÍSTICAS DE PRIMEFACES.	7
1.5.2 VERSIONES.....	7
1.6 WILDFLY	8
1.6.1 SERVIDOR DE APLICACIONES JBOSS	8
1.6.2 CARACTERÍSTICAS DE WILDFLY.....	8
1.7 MERCURIAL	10
1.7.1 CARACTERÍSTICAS DE MERCURIAL.....	10
1.8 BASE DE DATOS MYSQL.....	11

CAPÍTULO 2	12
ANÁLISIS Y DISEÑO DEL SISTEMA	12
2.1 INTRODUCCIÓN	12
2.2 ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE (ERS).....	12
2.2.1 FUNCIONES DEL SISTEMA	12
2.2.2 LEVANTAMIENTO DE REQUERIMIENTOS.....	14
2.3 DISEÑO DE DATOS.	31
2.3.1 DIAGRAMA ENTIDAD – RELACIÓN	31
2.3.2 DICCIONARIO DE DATOS	32
2.4 DISEÑO DE PROCESOS.	41
2.4.1 DIAGRAMA DE ACTIVIDADES.....	41
2.5 DISEÑO ARQUITECTÓNICO.....	46
2.5.1 DIAGRAMA DE DESPLIEGUE.....	46
2.6 DISEÑO DE INTERFACES.	47
2.6.1 DESCRIBIR ELEMENTOS DE LA INTERFAZ DEL SISTEMA.....	47
CAPÍTULO 3	55
DESARROLLO DE LA APLICACIÓN.....	55
3.1 INTRODUCCIÓN.	55
3.2 DESCRIPCIÓN DE LA ARQUITECTURA DEL SOFTWARE	55
3.3 CREACIÓN DE MANTENIMIENTOS INDISPENSABLES.	57
3.4 CREACIÓN DE LAS CLASES NECESARIAS PARA FORMAR EL XML DE LA FACTURA ELECTRÓNICA.....	57
3.4.1 INFORMACIÓN REQUERIDA PARA LA CONSTRUCCIÓN DEL XML.....	57
3.5 PROCESO DE FIRMA DEL COMPROBANTE ELECTRÓNICA.....	63
3.5.1 DESCRIPCIÓN DE CLASES QUE FIRMA EL XML.....	66
3.6 ESTRUCTURA XML DE UNA FACTURA ELECTRÓNICA.....	68
3.7 RIDE DE LA FACTURA ELECTRÓNICA.....	74
3.8 PRUEBAS DEL APLICATIVO	75
3.8.1 PRUEBAS DE FUNCIONAMIENTO BASADO EN CASOS DE USO ...	75
3.8.2 PRUEBAS DE INTEGRACIÓN.....	79
3.8.3 PRUEBAS DE RENDIMIENTO (TIEMPOS DE CONSULTA, HISTÓRICO)	79
CONCLUSIONES.....	81
RECOMENDACIONES.....	82
ANEXOS.....	83

ANEXO 1: INTEGRACIÓN DE LAS HERRAMIENTAS DE DESARROLLO	83
ANEXO 2: INSTALACIÓN DEL SERVIDOR DE APLICACIONES WILDFLY ...	84
ANEXO 3: INSTALACIÓN DEL ADMINISTRADOR DE PROYECTOS MAVEN	90
ANEXO 4: INSTALACIÓN DE HIBERNATE.....	92
ANEXO 5: INSTALACIÓN Y CONFIGURACIÓN DE LA BASE DE DATOS	94
BIBLIOGRAFÍA.....	95

ÍNDICE DE ILUSTRACIONES

IMAGEN 1: CLASE PROFESOR	3
IMAGEN 2: TABLA PROFESOR.....	3
IMAGEN 3: FUNCIONAMIENTO DE MVC.....	5
IMAGEN 4: CASO DE USO DEL USUARIO DEL SISTEMA.....	19
IMAGEN 5: CASO DE USO DE FUNCIÓN DE GESTIÓN DE PERSONA	20
IMAGEN 6: CASO DE USO DE GESTIÓN DE PERSONAS	20
IMAGEN 7: CASO DE USO DE GESTIÓN DE USUARIOS	21
IMAGEN 8: CASO DE USO DE GESTIÓN DE USUARIOS	21
IMAGEN 9: CASO DE USO DE GESTIÓN DE EMPRESA.....	22
IMAGEN 10: CASO DE USO DE GESTIÓN DE EMPRESAS.....	22
IMAGEN 11: CASO DE USO DE GESTIÓN DE ÍTEMS	23
IMAGEN 12: CASO DE USO DE GESTIÓN DE ÍTEMS	23
IMAGEN 13: CASO DE USO DE GESTIÓN DE CLIENTES	24
IMAGEN 14: CASO DE USO DE GESTIÓN DE CLIENTES	24
IMAGEN 15: CASO DE USO DE GESTIÓN DE TIPOS DE PERSONA	25
IMAGEN 16: CASO DE USO DE GESTIÓN DE TIPOS DE PERSONA	25
IMAGEN 17: CASO DE USO DE GESTIÓN DE TIPOS DE IDENTIFICACIÓN	26
IMAGEN 18: CASO DE USO DE GESTIÓN DE TIPOS DE IDENTIFICACIÓN	26
IMAGEN 19: CASO DE USO DE GESTIÓN DE TIPOS DE CONTRIBUYENTE.....	27
IMAGEN 20: CASO DE USO DE GESTIÓN DE TIPOS DE CONTRIBUYENTE.....	27
IMAGEN 21: CASO DE USO DE GESTIÓN DE IMPUESTOS	28
IMAGEN 22: CASO DE USO DE GESTIÓN DE IMPUESTOS	28
IMAGEN 23: CASO DE USO DE GESTIÓN DE CÓDIGOS DE IMPUESTOS	29
IMAGEN 24: CASO DE USO DE GESTIÓN DE CÓDIGOS DE IMPUESTOS	29
IMAGEN 25: CASO DE USO DE GESTIÓN DE EMISIÓN DE FACTURA ELECTRÓNICA	30
IMAGEN 26: CASO DE USO DE GESTIÓN DE EMISIÓN DE FACTURA ELECTRÓNICA	30
IMAGEN 27: DIAGRAMA ENTIDAD - RELACIÓN DEL PROYECTO.....	31
IMAGEN 28: DIAGRAMA DE SECUENCIA INICIO DE SESIÓN	42
IMAGEN 29: DIAGRAMA DE SECUENCIA DE GESTIÓN DE PERSONAS	42
IMAGEN 30: DIAGRAMA DE SECUENCIA DE GESTIÓN DE USUARIOS.....	43

IMAGEN 31: DIAGRAMA DE SECUENCIA DE GESTIÓN DE EMPRESAS	43
IMAGEN 32: DIAGRAMA DE SECUENCIA DE GESTIÓN DE ÍTEMS.....	44
IMAGEN 33: DIAGRAMA DE SECUENCIA DE GESTIÓN DE CLIENTES.....	44
IMAGEN 34: DIAGRAMA EMISIÓN DE FACTURA ELECTRÓNICA	45
IMAGEN 35: DIAGRAMA DE DESPLIEGUE.....	46
IMAGEN 36: PANTALLA PRINCIPAL DEL SISTEMA.....	47
IMAGEN 37: PANTALLA DE LA CABECERA DE LA PANTALLA	48
IMAGEN 38: MENÚ DEL SISTEMA.....	48
IMAGEN 39: CUERPO DEL SISTEMA, LUGAR DONDE SE ABREN TODAS LAS PANTALLAS	49
IMAGEN 40: FORMATO DE LISTA DE REGISTROS	50
IMAGEN 41: PANTALLA DE VISUALIZACIÓN DE IMÁGENES	50
IMAGEN 42: PANTALLA DE MODIFICACIÓN DE DATOS	51
IMAGEN 43: DIALOG PARA MODIFICACIÓN DE DATOS.....	51
IMAGEN 44: DIALOG PARA ELIMINACIÓN DE REGISTROS	52
IMAGEN 45: FILTROS DE BÚSQUEDA	53
IMAGEN 46: FORMATO DE BOTONES.....	53
IMAGEN 47: FORMATO DE PANTALLA PARA EL MANTENIMIENTO DE DATOS .	54
IMAGEN 48: DESCRIPCIÓN DE CARACTERES QUE FORMAN PARTE DE LA CLAVE DE ACCESO	58
IMAGEN 49: TIPO DE EMISIÓN DEL COMPROBANTE.....	58
IMAGEN 50: COMPROBANTES QUE SE PUEDE EMITIR ELECTRÓNICAMENTE....	59
IMAGEN 51: TIPOS DE AMBIENTES UTILIZADOS PARA LA EMISIÓN	59
IMAGEN 52: TIPOS DE IDENTIFICACIÓN QUE SE PUEDE UTILIZAR PARA EL CLIENTE	60
IMAGEN 53: CAMPOS QUE FORMAN PARTE DE LA CLAVE DE ACCESO.....	60
IMAGEN 54: FORMATO DE CALCULAR EL DÍGITO VERIFICADOR MEDIANTE EL MÓDULO 11	62
IMAGEN 55: CÓDIGOS DE IMPUESTOS	62
IMAGEN 56: PORCENTAJES DE IMPUESTOS	63
IMAGEN 57: FORMATO DE FIRMA ELECTRÓNICA	63
IMAGEN 58: RIDE O PDF DE LA FACTURA ELECTRÓNICA.....	74
IMAGEN 59: PANTALLA PRINCIPAL DEL SISTEMA.....	76
IMAGEN 60: ELEMENTOS QUE FORMAN PARTE DEL SISTEMA	76
IMAGEN 61: FORMATO DE LA PANTALLA PRINCIPAL DE LA FACTURA ELECTRÓNICA	77
IMAGEN 62: FORMATO DE MENSAJES DE NOTIFICACIONES	78

IMAGEN 63: FORMATO DE MENSAJES DE NOTIFICACIÓN DE ERRORES	78
IMAGEN 64: MENSAJE DE INDISPONIBILIDAD DEL SRI	80
IMAGEN 65: PANTA BIENVENIDA WILDFLY	89
IMAGEN 66: ECLIPSE MARKETPLACE - ECLIPSE MARS.2 RELEASE (4.5.2)	93

ÍNDICE DE TABLAS

TABLA 1: GESTIÓN DE PERSONAS	14
TABLA 2: FUNCIÓN DE GESTIÓN DE USUARIOS	14
TABLA 3: FUNCIÓN DE GESTIÓN DE EMPRESAS	15
TABLA 4: FUNCIÓN DE GESTIÓN DE ÍTEMS.....	15
TABLA 5: FUNCIÓN DE GESTIÓN DE CLIENTES.....	16
TABLA 6: FUNCIÓN DE GESTIÓN DE TIPOS DE PERSONA.....	16
TABLA 7: FUNCIÓN DE GESTIÓN DE IDENTIFICACIÓN.....	16
TABLA 8: FUNCIÓN DE GESTIÓN DE TIPOS DE CONTRIBUYENTE	17
TABLA 9: FUNCIÓN DE GESTIÓN DE IMPUESTOS.....	17
TABLA 10: FUNCIÓN DE GESTIÓN DE IMPUESTOS.....	18
TABLA 11: FUNCIÓN DE EMISIÓN DE FACTURACIÓN ELECTRÓNICA	18
TABLA 12: FUNCIÓN DE GESTIÓN DE PERSONAS	20
TABLA 13: FUNCIÓN DE GESTIÓN DE USUARIOS	21
TABLA 14: FUNCIÓN DE GESTIÓN DE EMPRESAS	22
TABLA 15: FUNCIÓN DE GESTIÓN DE ÍTEMS.....	23
TABLA 16: FUNCIÓN DE GESTIÓN DE CLIENTES.....	24
TABLA 17: FUNCIÓN DE GESTIÓN DE TIPOS DE PERSONA.....	25
TABLA 18: FUNCIÓN DE GESTIÓN DE TIPOS DE IDENTIFICACIÓN.....	26
TABLA 19: FUNCIÓN DE GESTIÓN DE TIPOS DE CONTRIBUYENTE	27
TABLA 20: FUNCIÓN DE GESTIÓN DE IMPUESTOS.....	28
TABLA 21: FUNCIÓN DE GESTIÓN DE CÓDIGOS DE IMPUESTOS.....	29
TABLA 22: FUNCIÓN DE GESTIÓN DE FACTURA ELECTRÓNICA.....	30
TABLA 23: LISTADO DE TABLAS DEL MODELO ENTIDAD - RELACIÓN	32
TABLA 24: LISTADO DE REFERENCIAS PRIMARIAS	33
TABLA 25: LISTADO DE REFERENCIAS FORÁNEAS.....	34
TABLA 26: LISTADO DE CAMPOS DE LA TABLA TCFACTURA	35
TABLA 27: LISTADO DE CAMPOS DE LA TABLA TCLIENTES.....	35
TABLA 28: LISTADO DE CAMPOS DE LA TABLA TCODIGOSIMPUESTOS.....	36
TABLA 29: LISTADO DE CAMPOS DE LA TABLA TIMPUUESTOS	36
TABLA 30: LISTADO DE CAMPOS DE LA TABLA TDFACTURA.....	37
TABLA 31: LISTADO DE CAMPOS DE LA TABLA TDFACTURAIMPUUESTOS.....	37
TABLA 32: LISTADO DE CAMPOS DE LA TABLA TEMPRESAS.....	38

TABLA 33: LISTADO DE CAMPOS DE LA TABLA TITEMS	38
TABLA 34: LISTADO DE CAMPOS DE LA TABLA TITEMSIMPUESTOS	38
TABLA 35: LISTADO DE CAMPOS DE LA TABLA TPERSONAS	39
TABLA 36: LISTADO DE CAMPOS DE LA TABLA TTIPOSCONTRIBUYENTE	40
TABLA 37: LISTADO DE CAMPOS DE LA TABLA TTIPOSIDENTIFICACION	40
TABLA 38: LISTADO DE CAMPOS DE LA TABLA TTIPOSPERSONA	40
TABLA 39: LISTADO DE CAMPOS DE LA TABLA TUSUARIOS.....	41
TABLA 40: NÚMERO DE MÓDULOS UTILIZADOS EN EL SISTEMA	56
TABLA 41: NÚMERO DE TRANSACCIONES UTILIZADAS.....	56
TABLA 42: ESTRUCTURA XML DE UNA FACTURA ELECTRÓNICA	73
TABLA 43: TECNOLOGÍAS DISPONIBLES EN LA VERSIÓN 10.0.0 DE WILDFLY. ..	86
TABLA 44: ESTRUCTURA DE DIRECTORIOS WILDFLY 10.	88
TABLA 45: REQUISITOS PARA INSTALACIÓN MAVEN 3.3.9.....	90

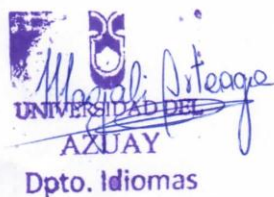
RESUMEN

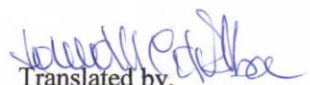
El propósito de este Proyecto de Titulación consiste en elaborar una aplicación para emitir facturas electrónicas, utilizando Maven, Hibernate, Java Server Faces (JSF), Prime Faces y MySQL como elementos del entorno de desarrollo, permitiéndonos integrar, gestionar y administrar de mejor manera nuestros proyectos. Se hace énfasis en Hibernate y Maven porque son consideradas como dos de las mejores herramientas para desarrollo software con Java y a la vez, son poco conocidas en nuestro medio.

De esta manera se intenta solventar los problemas que se presentan en el desarrollo de software, como son: la integración de módulos, la gestión de proyectos, el control de versiones y la desorganización en la creación de código fuente. Además con esta aplicación se intenta contribuir para que las empresas que deben cumplir el esquema de emisión de comprobantes electrónicos definido por el Servicio de Rentas Internas, lo hagan con un sistema eficiente y eficaz.

ABSTRACT

This graduation project aims to develop an application to issue electronic invoices, using Maven, Hibernate, Java Server Faces (JSF), PrimeFaces and MySQL as components of development environment; enabling us to integrate, manage and administer our projects in the best possible way. Special emphasis is made in Hibernate and Maven because they are considered two of the best tools for software development with Java, and at the same time, are little known in our environment. Thus, the aim is to solve the problems arising in software development, such as: module integration, project management, version control and disorganization in the creation of source code. In addition, we seek to contribute to the companies that must comply with the scheme of issuing electronic vouchers defined by the Internal Revenue Service, to carry out this work efficiently and effectively through the use of this application




Translated by,
Lic. Lourdes Crespo

CAPÍTULO 1

TECNOLOGÍAS APLICADAS

1.1 Introducción

En este capítulo se describirán las herramientas Maven, Hibernate, Java Server Faces (JSF), Prime Faces y MySQL, que en conjunto constituyen el entorno de desarrollo que ha permitido la integración, gestión y administración eficiente de este proyecto.

1.2 Maven

Es una herramienta de software para la gestión y construcción de proyectos Java que posee objetivos predefinidos para realizar tareas como la compilación del código y su empaquetado. Es una forma coherente de organización de grandes colecciones de módulos interdependientes y bibliotecas, que hacen uso de decenas o cientos de componentes de terceros, lo cual facilita el trabajo de los programadores y permite a las organizaciones evolucionar en la gestión de proyecto

Maven se centra en el pre-procesamiento, compilación, empaquetado, prueba y distribución de aplicaciones. Básicamente es una herramienta de gestión de proyectos, que permite compilar, empaquetar, generar documentación, ejecutar pruebas, preparar builds, etc. También se pueden generar informes, sitios web y facilita la comunicación entre los miembros de un equipo de trabajo

Con esta herramienta se pueden usar dependencias y librerías (propias y de terceros) añadiéndolas al POM (Project Object Model - Modelo de proyecto de Objetos), que es donde la identidad y la estructura del proyecto es declarada

Ejemplo:

```
<dependency>
    <groupId>org.primefaces</groupId>
    <artifactId>primefaces</artifactId>
    <version>5.3</version>
</dependency>
```

El fichero pom.xml define un proyecto Maven, de forma similar a la configuración descrita por el web.xml en aplicaciones web. Este archivo constituye el mapa figurativo que interpreta Maven.

En el ejemplo se agrega una librería Primefaces para usar Maven en un proyecto. Esta herramienta además gestiona otras características como el manejo de las licencias, gestión y aportes de usuarios, dependencia con otros proyectos, etc.

Ventajas:

Maven se basa en patrones y en estándares. Esto permite a los desarrolladores moverse entre proyectos facilitando la compilación o el empaquetado. Esto mejora el mantenimiento y la reusabilidad.

Maven mejora la gestión de librerías, incluso teniendo en cuenta las dependencias transitivas, por ejemplo, si A depende de B y B depende de C, sin que exista dependencia directa entre A y C, Maven añadirá las librerías B y C en el paquete.

1.3 Hibernate

Hibernate es una herramienta de Mapeo objeto-relacional (ORM). ORM es simplemente el código requerido para guardar el valor de nuestras clases en una base de datos relacional, es decir, es un framework de persistencia de los datos. Los objetos generados se conocen como beans o persistencias.

En 2001, Gavin King creó Hibernate, no se basó en ninguna Solicitud de Especificaciones Java (JSR), ni en ninguna especificación, sino creó un framework de ORM exitoso y de amplia difusión.

1.3.1 Características Hibernate:

- No depende de estándares, lo que hace que pueda añadir funcionalidades más rápidamente.
- Al igual que el API de Java Data Objects (JDO), no es necesario implementar interfaces o heredar de clases.
- Maven sólo puede persistir a bases de datos relacionales, a diferencia de JDO que permite otro tipo de repositorios.

Ejemplo de ORM:

En el siguiente diagrama UML observamos los atributos de la clase Profesor

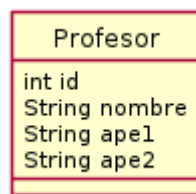


Imagen 1: Clase Profesor

La siguiente tabla contiene los campos necesarios para almacenar los objetos de la clase Profesor en la tabla PROFESOR de la base de datos.

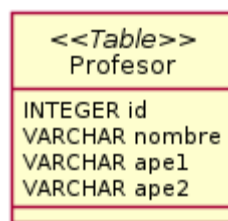


Imagen 2: Tabla Profesor

Se conoce como Mapeo - Objeto - Relacional al código Java que permite a los atributos del objeto Profesor almacenar, borrar o actualizar un registro en la tabla PROFESOR.

1.3.2 Comparación entre el API de persistencias de Java (JPA) e Hibernate

A diferencia de JPA, Hibernate implementa como parte de su código la especificación de JPA, es decir, se puede crear una capa de persistencia utilizando Hibernate apoyándonos en las definiciones y reglas que se especifican en el API de Persistencia de Java, aunque esto no es un requisito indispensable.

Hibernate ofrece más funcionalidades que la implementación estándar JPA, por ejemplo, es capaz trabajar con diferentes bases de datos noSQL, algo que JPA no cubre. Esta funcionalidad nos permite utilizar bases de datos orientadas a documentos, por ejemplo Mongo DB.

Para nuestro proyecto de tesis optamos por Hibernate por su facilidad de mapeo de tablas y la rapidez con la que nos permite el desarrollo de los objetos Java Planos, haciéndola una herramienta muy potente que simplifica el desarrollo de proyectos Maven o proyectos web basados en java.

1.4 Java Server Faces JSF

Java Server Faces es un framework para desarrollo de interfaces de usuario en aplicaciones Web complejas basadas en Java. Está incluido en la plataforma Java Enterprise Edition (JEE), lo cual permite desarrollar aplicaciones sin incluir ninguna librería adicional en el proyecto. Además cuenta con un soporte completo que le permite trabajar con distintos frameworks web independientes.

1.4.1 ¿Qué es Java Server Faces?

“JavaServer™ Faces (JSF) es el marco estándar de interfaz de usuario orientada a componentes (UI) para la plataforma Java Enterprise Edition JEE. Es también conocido como un framework de desarrollo web basado en Java.” (JavaServer™)

1.4.2 Características de Java Server Faces

➤ Trabaja con MVC (Modelo – Vista - Controlador).- Utiliza unos de los patrones de arquitectura de software más conocidos en el desarrollo web modelo-vista-controlador (MVC). Esta arquitectura propone la construcción de tres componentes:

- Modelo. Es el componente que se encarga de los datos o representación de la información.
- Vista. Este componente se encarga de la interacción con el usuario.
- Controlador. Este último componente enlaza la vista con el modelo.

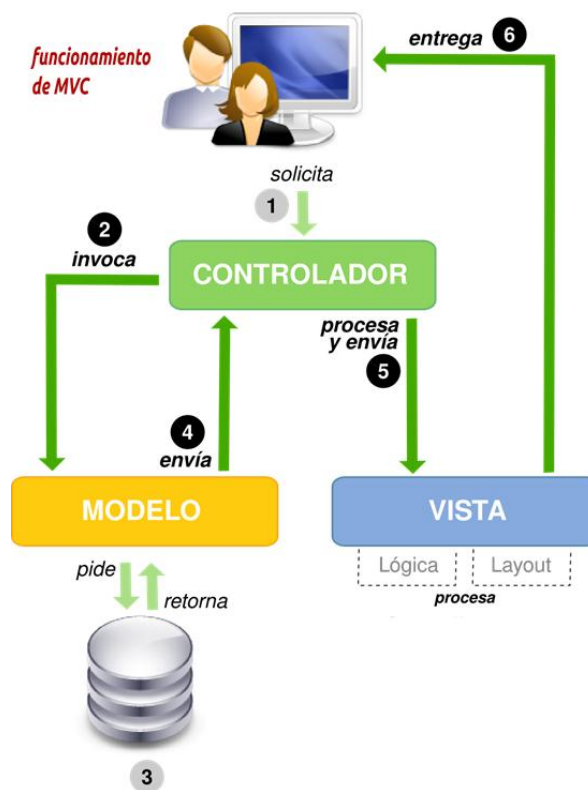


Imagen 3: Funcionamiento de MVC.

- Desarrollo Rápido de Aplicaciones (RAD). Permite desarrollar, de una manera rápida, un software con bajo costo de mantenimiento y ágil desarrollo de procesos nuevos.
- Posee un conjunto de componentes de interfaz de usuario listos para usarse.
- Permite la utilización de JavaScript en la página para mejorar la interacción con la interfaz del usuario.
- Es posible elaborar diferentes componentes con requerimientos específicos.
- Su funcionamiento puede ser controlado mediante APIs
- Tiene un modelo de eventos en el lado del servidor.

1.4.3 Versiones

- Java Server Faces 1.0.- Su lanzamiento inicia el 11 de marzo de 2004 con las especificaciones de JSF.
- Java Server Faces 1.1. – Con el lanzamiento de esta versión el 27 de mayo de 2004 se logra solucionar varios errores.
- Java Server Faces 1.2.- El 11 de mayo de 2006 se presentan mejoras en las funciones y corrección de errores.
- Java Server Faces 2.0.- El 12 de agosto de 2009 el framework mejora su funcionalidad, rendimiento y es fácil de utilizar.
- Java Server Faces 2.1.- Esta versión lanzada el 22 de octubre de 2010 no posee mayores cambios y mejora su mantenimiento.
- Java Server Faces 2.2.- Esta versión del 16 de abril de 2013 brinda soporte a HTML 5, vistas sin estado y contratos con librerías de recursos.

1.5 Prime Faces

Es una suite open source de componentes que extiende a Java Server Faces. “Prime Faces es una librería de componentes visuales open source desarrollada y mantenida por Prime Technology, una compañía Turca de IT especializada en consultoría ágil, JSF y Java Enterprise Edition” (Lerma, Adictos al trabajo, 2015).

Prime Faces es un framework de código abierto muy popular para Java Server Faces. Posee un kit optimizado para desarrollo de aplicaciones móviles con un gran número de validadores que nos facilita la verificación del lado del cliente. También incluye temas prediseñados los cuales podemos administrar según requisitos y necesidades del cliente, además de un conjunto de más de 100 componentes entre los que tenemos un editor de HTML, autocompletar, cartas, gráficas o paneles, entre otros.

1.5.1 Características de Primefaces.

- Tiene un gran conjunto de componentes UI (editor html, cuadros de dialogo, auto completar, gráficos, selectores de fechas, tablas de datos, componentes de árbol, etc).
- Soporte nativo de Ajax fundamentada en la API Ajax JSF 2.0.
- Posee un conjunto de herramientas que permite elaborar aplicaciones web para dispositivos móviles.
- Se puede usar JavaScript directamente dentro de una sección `<script>`.
- Esta librería está empaquetada dentro de un archivo Jar por tanto es muy ligera.
- No es necesario realizar ninguna configuración.
- Tiene más de 35 temas pre construidos.
- Es posible manejar un editor de temas.

1.5.2 Versiones

- Primefaces 1: Trabaja con JSF 1.2
- Primefaces 2: Trabaja con JSF 2

1.6 WildFly

“WildFly, anteriormente conocido como JBoss, es un servidor de aplicaciones Java EE de código abierto implementado en Java puro. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo para el que esté disponible la máquina virtual de Java.” (Red Hat, 2016)

1.6.1 Servidor de aplicaciones JBoss

JBoss, es un servidor de aplicaciones de código abierto basado en estándares J2EE 1.4, implementado al 100% en Java. Proporciona servicios que soportan la ejecución y disponibilidad de las aplicaciones, define una arquitectura para la administración de recursos, acceso a datos y persistencia, entre otros.

JBoss puede ser utilizado fácilmente y se puede descargar de manera gratuita ya que posee una licencia de código abierto GNU que posibilita su distribución sin ninguna restricción.

1.6.2 Características de WildFly

- Servidor de aplicaciones que posee una licencia GNU de código abierto que no tiene ningún precio adicional.
- Estándares soportados
 - Portlet Specification and API 1.0 (JSR-168)
 - Content Repository for Java Technology API (JSR-170)
 - Java Server Faces 2.0 (JSR-252)

- Java Management Extensión (JMX) 1.2
 - Compatibilidad 100% con J2EE 1.4 al utilizar JBoss AS.
- A nivel empresarial ofrece un alto nivel de confiabilidad.
 - Incrustable, arquitectura orientada de servicios.
 - Flexibilidad sólida.
 - Ofrece interacción con otras aplicaciones a cualquier objeto de Java.
 - Brinda asistencia completa para Java Management Extensions – JMX
 - Implementa la especificación inicial de EJB 3.0. Enterprise JavaBean.- Los EJB escritos en lenguaje de programación Java suministran un modelo de mecanismos distribuidos estándar del lado del servidor, que encapsula la lógica de negocio de una aplicación. Una característica importante de los Java Beans Empresariales es proporcionar al desarrollador un modelo que le permita mitigar los principales inconvenientes de una aplicación institucional (conurrencia a datos, transacciones, persistencia, seguridad, etc.) para concentrarse en resolver los problemas de la lógica empresarial. Permiten su reutilización debido a que se fundamentan en componentes brindando de esta manera una mayor flexibilidad.
 - Está orientado a trabajar con Programación Orientada a Aspectos (AOP). Este paradigma de programación nos permite encapsular diferentes conceptos que componen una aplicación eliminando las preocupaciones colaterales como: patrones de diseño, comprobación de errores, manejo de restricciones en la sincronización y optimizaciones de rendimiento. Es fácil de aprender y usar, compatible con la plataforma Java y disponible de forma gratuita, incluso para su uso en el desarrollo de productos empresariales.
 - Trabaja con Hibernate.- Esto permite realizar el correspondiente mapeo de objetos Java a las tablas de la base de datos, lo cual facilita a los programadores de Java, crear las clases de persistencia. Además Hibernate brinda servicios de consulta y recuperación de información.

- Posee JBoss Cache.- Ofrece soluciones para almacenar en memoria los objetos Java a los cuales se acceden frecuentemente, reduciendo así los accesos innecesarios a la base de datos, proporcionando soluciones avanzadas de clustering a nivel empresarial y mejorando el rendimiento.

1.7 Mercurial

Es una herramienta open source de gestión de control de código fuente distribuido que nos brinda la posibilidad de manejar de manera eficiente proyectos de cualquier tamaño. Posee un sistema de control de versiones multiplataforma para desarrolladores de software. Está implementado principalmente haciendo uso del lenguaje de programación Python, pero incluye una implementación binaria de diff escrita en C.

Mercurial fue escrito originalmente para funcionar sobre GNU/Linux. Pero ha sido adaptado para Windows, Mac OS X y la mayoría de otros sistemas tipo Unix. Mercurial es, sobre todo, un programa para línea de comandos. Todas las operaciones de Mercurial se invocan como opciones dadas a su programa motor, hg (cuyo nombre hace referencia al símbolo químico del mercurio).

Las principales metas de desarrollo de Mercurial incluyen:

- Mejoras en el rendimiento y escalabilidad.
- Desarrollo completamente distribuido, sin necesidad de un servidor.
- Gestión robusta de archivos, tanto de texto como binarios.
- Capacidades avanzadas de ramificación e integración.
- Sencillez conceptual.
- Interfaz web integrada.

1.7.1 Características de Mercurial

- Es rápido y potente.- Gestiona de manera eficaz los proyectos de cualquier tamaño y tipo. Cada clon contiene toda la historia del proyecto, por lo que la

mayoría de las acciones son locales, rápidas y cómodas. Mercurial es compatible con una multitud de flujos de trabajo y se puede incrementar su funcionalidad con extensiones.

- Es fácil de aprender.- Podemos crear un proyecto con muy pocas instrucciones y comenzar a trabajar de inmediato. Por ejemplo, los comandos para crear un proyecto son:

```
$ hg init (project-directory)
$ cd (project-directory)
$ (add some files)
$ hg add
$ hg commit -m 'Initial commit'
```

1.8 Base de Datos MySQL.

Es el administrador de bases de datos de código abierto más popular del mundo basado en SQL (Lenguaje de Consulta Estructurado). MySQL es multiplataforma lo cual permite ejecutarse en prácticamente todas las plataformas (Linux, Windows, Mac, etc.).

“MySQL es un software de código abierto licenciado, aunque MySQL AB distribuye una versión comercial que se diferencia de la versión libre sólo en el soporte técnico que ofrece y en la posibilidad de integrar este gestor en un software propietario.

El lenguaje de programación que utiliza MySQL es Structured Query Language (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales”. (Group, 2016)

CAPÍTULO 2

ANÁLISIS Y DISEÑO DEL SISTEMA

2.1 Introducción

En este capítulo describiremos los requerimientos del sistema mediante la utilización de diferentes casos de uso. Se establece el diseño de datos mediante el modelo entidad relación y su correspondiente diccionario de datos. Se visualizarán los diagramas de actividad en el desarrollo de procesos y los diagramas de despliegue correspondiente al diseño arquitectónico que posee el sistema. Al final se describe el diseño de interfaz del sistema el cual permite una adecuada interacción con el usuario.

2.2 Especificación de Requisitos de Software (ERS)

Representación del comportamiento del sistema de facturación electrónica, describiendo los requerimientos y construyendo un conjunto de casos de uso que representan las interacciones que tendrán los usuarios con la aplicación.

2.2.1 Funciones del Sistema

Este sistema proporciona las siguientes funciones que podemos clasificar en:

a) Almacenamiento de datos

- Datos de la empresa emisora de la facturación electrónica
- Personas relacionadas con la facturación electrónica (Usuarios, Empresa emisora, Clientes)
- Tipos de persona (Natural, Jurídica)
- Tipos de identificación (Cédula, Pasaporte, Registro Único de Contribuyentes, etc.)

- Tipos de contribuyente
- Usuarios que manipulen el sistema de Facturación Electrónica
- Ítems que se usarán para el proceso de facturación.
- Información de la factura Electrónica

b) Mantenimientos

- Mantenimiento de personas
- Mantenimiento de empresas
- Mantenimiento de usuarios
- Mantenimiento de ítems
- Mantenimiento de tipos de persona
- Mantenimiento de tipos de identificación
- Mantenimiento de tipos de contribuyente
- Mantenimiento de clientes
- Mantenimiento de impuestos
- Mantenimiento de códigos de impuestos
- Emisión de Facturación Electrónica

c) Consultas y reportes

- Consulta de personas
- Consulta de empresas
- Consulta de usuarios
- Consulta de ítems
- Consulta de tipos de persona
- Consulta de tipos de identificación
- Consulta de tipos de contribuyente
- Consulta de clientes
- Consulta de impuestos
- Consulta de códigos de impuestos
- Reporte listado de personas
- Reporte listado de ítems
- Reporte listado de clientes
- RIDE Facturación Electrónica

d) Procesos

- RIDE Facturación Electrónica (generación y autorización)

2.2.2 Levantamiento de requerimientos

2.2.2.1 Requerimientos

Ref. #	Función de Gestión de Personas
R1.1	Registrar Personas. Ingreso de la información básica de una persona que luego será registrada como Usuario, Empresa emisora o Cliente.
R1.2	Modificar Personas. El usuario con rol de administrador podrá modificar la información de una persona.
R1.3	Consultar Personas. Los usuarios con permisos a la transacción de consulta, podrán consultar la información de las personas.
R1.4	Eliminar Persona. Un usuario administrador podrá eliminar la persona deseada.

Tabla 1: Gestión de Personas

Ref. #	Función de Gestión de Usuarios
R2.1	Registro de Usuario. El usuario ingresará sus datos de usuario y contraseña para que el sistema lleve un registro del mismo.
R2.2	Modificar Usuario. El usuario accederá al sistema para modificar sus datos, como puede ser su contraseña.
R2.3	Consultar datos del Usuario. Permite consultar los datos de un usuario.
R2.4	Eliminar Usuario. Un usuario administrador podrá eliminar un usuario.

Tabla 2: Función de Gestión de Usuarios

Ref. #	Función de Gestión de Empresas
R3.1	Registro de Empresa. Ingreso de la información de la empresa emisora de la facturación electrónica.
R3.2	Modificar Empresa. Modificación de la información de la empresa.
R3.3	Consultar Empresa. Consulta de la información de la empresa
R3.4	Eliminar Empresa. Un usuario administrador podrá eliminar el registro de la empresa.

Tabla 3: Función de Gestión de Empresas

Ref. #	Función de Gestión de Ítems
R4.1	Ingreso de Ítems. El usuario encargado de la transacción de ítems podrá ingresar nuevos registros
R4.2	Modificar Ítem. Permite modificar la información del ítem.
R4.3	Consultar Ítem. Permite consultar información del ítem.
R4.4	Eliminar Ítem. El usuario encargado de la transacción de ítems podrá eliminar registros.

Tabla 4: Función de Gestión de Ítems

Ref. #	Función de Gestión de Clientes
R5.1	Registro de clientes. El usuario con los permisos adecuados podrá registrar nuevos cliente.
R5.2	Modificar Cliente. Permite modificar la información del cliente.
R5.3	Consultar Cliente. Permite consultar información del cliente.
R5.4	Eliminar Cliente. El usuario con los permisos adecuados

	podrá eliminar clientes.
--	--------------------------

Tabla 5: Función de Gestión de Clientes

Ref. #	Función de Gestión de Tipos de Persona
R6.1	Ingreso de tipos de persona. Permite ingresar los tipos de persona (Natural, Jurídica).
R6.2	Modificar tipo de personas. Permite modificar la información del tipo de persona.
R6.3	Consultar tipo de persona. Permite consultar información del tipo de persona.
R6.4	Eliminar tipo de persona. El usuario con los permisos adecuados podrá eliminar el tipo de persona seleccionado.

Tabla 6: Función de Gestión de Tipos de Persona

Ref. #	Función de Gestión de Tipos de Identificación
R7.1	Ingreso de tipos de identificación. Permite ingresar los tipos de identificación (Cédula, Ruc, Pasaporte, etc.).
R7.2	Modificar tipo de identificación. Permite modificar la información del tipo de identificación.
R7.3	Consultar tipo de identificación. Permite consultar información del tipo de identificación.
R7.4	Eliminar tipo de identificación. El usuario con los permisos adecuados podrá eliminar el tipo de identificación.

Tabla 7: Función de Gestión de Identificación

Ref. #	Función de Gestión de Tipos de Contribuyente
R8.1	Ingreso de tipos de contribuyentes. Permite ingresar los tipos

	contribuyente.
R8.2	Modificar tipo de contribuyentes. Permite modificar la información del tipo de contribuyente.
R8.3	Consultar tipo de contribuyentes. Permite consultar información del tipo de contribuyente.
R8.4	Eliminar tipo de contribuyentes. El usuario con los permisos adecuados podrá eliminar el tipo de contribuyente seleccionado.

Tabla 8: Función de Gestión de Tipos de Contribuyente

Ref. #	Función de Gestión de impuestos
R9.1	Ingreso de impuestos. Permite ingresar los diferentes tipos de impuestos, IVA, ICE.
R9.2	Modificar impuesto. Permite modificar la información de impuestos
R9.3	Consultar impuestos. Permite consultar información de los tipos de impuesto
R9.4	Eliminar impuesto. El usuario con los permisos adecuados podrá eliminar el impuesto deseado.

Tabla 9: Función de Gestión de Impuestos

Ref. #	Función de Gestión de Códigos de Impuestos
R10.1	Ingreso de códigos de impuestos. Permite ingresar la información de los códigos de los impuestos.
R10.2	Modificar códigos de impuestos. Permite modificar la información de los códigos de impuestos
R10.3	Consultar los códigos de impuestos. Permite consultar

	información de los códigos de impuestos
R10.4	Eliminar códigos de impuestos. El usuario con los permisos adecuados podrá eliminar los códigos de impuestos

Tabla 10: Función de Gestión de Impuestos

Ref. #	Función de Emisión de Facturación Electrónica
R11.1	Seleccionar cliente. Ingresar información necesaria del cliente.
R11.2	Validar información.- Validación de información requerida del cliente para la facturación electrónicas
R11.3	Ingreso Ítems.- El usuario ingresa el código, y la cantidad de los ítems solicitados por el cliente.
R11.4	Emitir Facturación Electrónica.

Tabla 11: Función de Emisión de Facturación Electrónica

2.2.2.2 Casos de uso de alto nivel

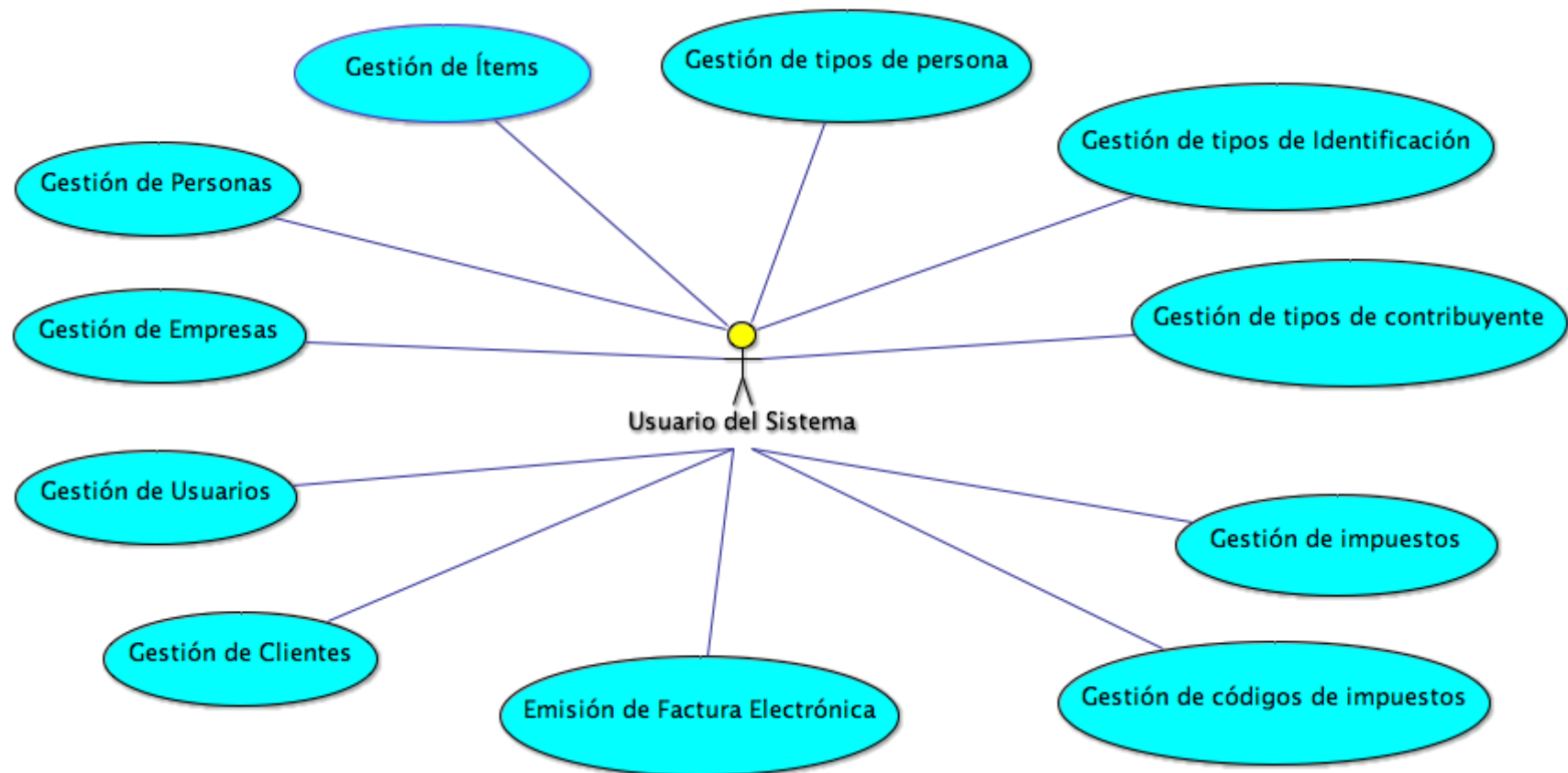


Imagen 4: Caso de Uso del Usuario del Sistema

➤ **Caso de uso de Función de Gestión de Personas**

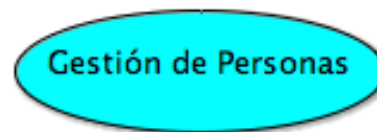


Imagen 5: Caso de Uso de Función de Gestión de Persona

Casos de uso: 1	Función de Gestión de Personas
Actor:	Usuarios
Propósito:	Ingreso de la información básica de una persona que luego será registrada como Usuario, Empresa emisora o Cliente.
Tipo:	Esencial
Referencia:	R.1.1. Registrar persona. R.1.2. Modificar personas. R.1.3. Consultar personas. R.1.4. Eliminar personas.

Tabla 12: Función de Gestión de Personas

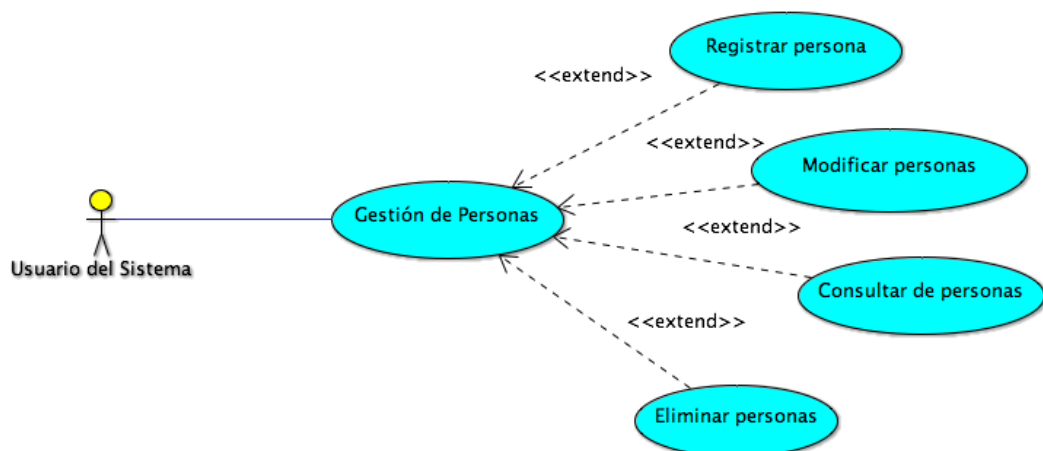


Imagen 6: Caso de Uso de Gestión de Personas

➤ **Caso de uso de Función de Gestión de Usuarios**



Imagen 7: Caso de Uso de Gestión de Usuarios

Casos de uso: 2	Función de Gestión de Usuarios
Actor:	Usuarios
Propósito:	Ingreso de los usuarios que utilizaran los diferentes módulos del sistema
Tipo:	Esencial
Referencia:	R.1.1. Registrar usuario. R.1.2. Modificar usuario. R.1.3. Consultar usuario. R.1.4. Eliminar usuario.

Tabla 13: Función de Gestión de Usuarios

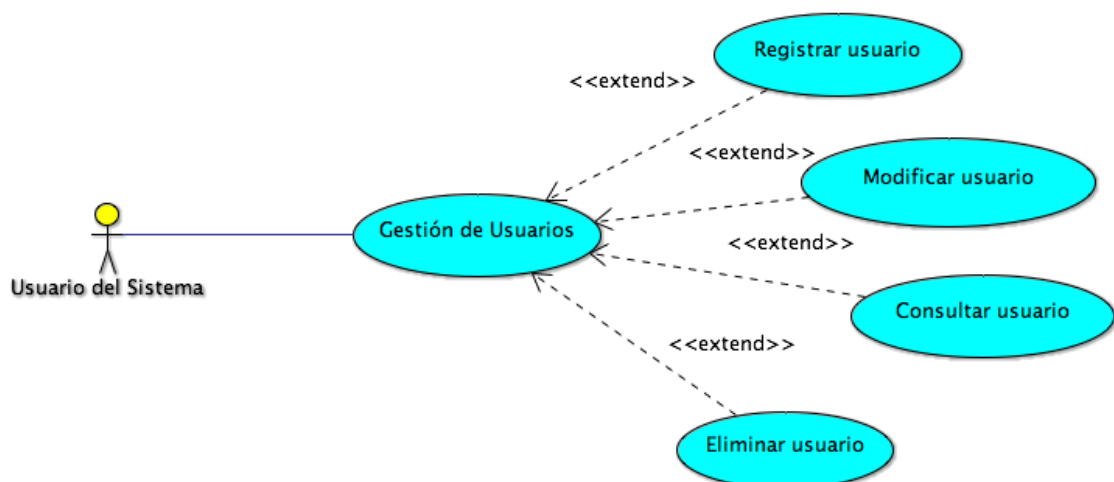


Imagen 8: Caso de Uso de Gestión de Usuarios

➤ **Caso de uso de Función de Gestión de Empresas**



Imagen 9: Caso de Uso de Gestión de Empresa

Casos de uso: 3	Función de Gestión de Empresas
Actor:	Usuarios
Propósito:	Gestionar la información de las empresas
Tipo:	Esencial
Referencia:	R.1.1. Registrar empresa. R.1.2. Modificar empresa. R.1.3. Consultar empresa. R.1.4. Eliminar empresa.

Tabla 14: Función de Gestión de Empresas

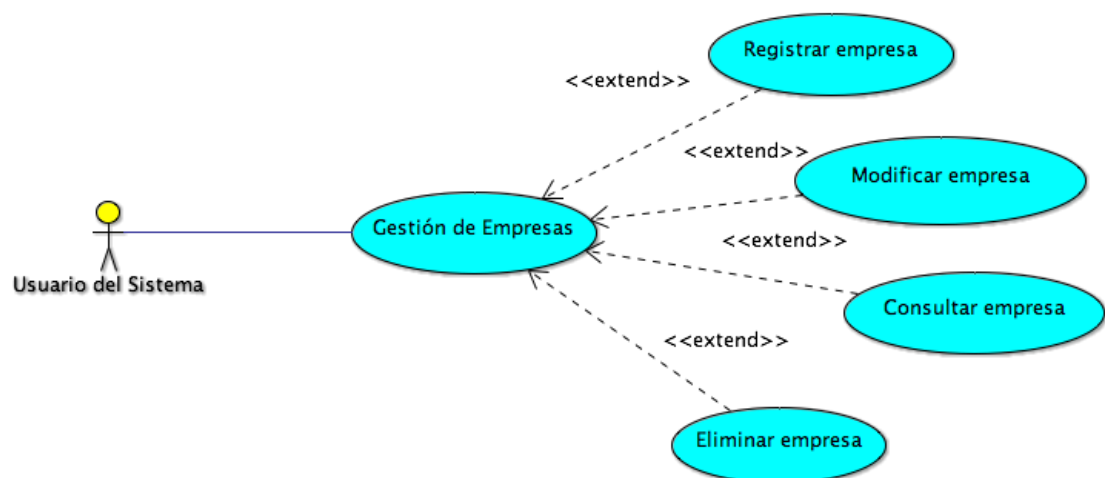


Imagen 10: Caso de Uso de Gestión de Empresas

➤ Caso de uso de Función de Gestión de Ítems



Imagen 11: Caso de Uso de Gestión de Ítems

Casos de uso: 4	Función de Gestión de Ítems
Actor:	Usuarios
Propósito:	Ingreso y mantenimiento de los ítems que se utilizaran en la facturación electrónica.
Tipo:	Esencial
Referencia:	R.1.1. Registrar ítem. R.1.2. Modificar ítem. R.1.3. Consultar ítem. R.1.4. Eliminar ítem.

Tabla 15: Función de Gestión de Ítems

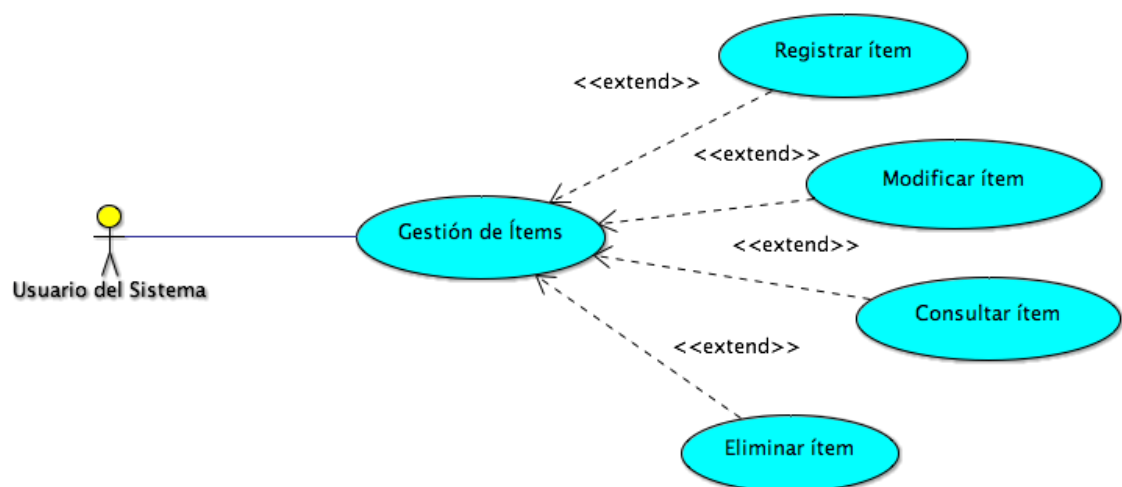


Imagen 12: Caso de Uso de Gestión de Ítems

➤ **Caso de uso de Función de Gestión de Clientes**



Imagen 13: Caso de Uso de Gestión de Clientes

Casos de uso: 5	Función de Gestión de Clientes
Actor:	Usuarios
Propósito:	Ingreso y mantenimiento de los clientes que se utilizaran en la emisión de la facturación.
Tipo:	Esencial
Referencia:	R.1.1. Registrar cliente. R.1.2. Modificar cliente. R.1.3. Consultar cliente. R.1.4. Eliminar cliente.

Tabla 16: Función de Gestión de Clientes

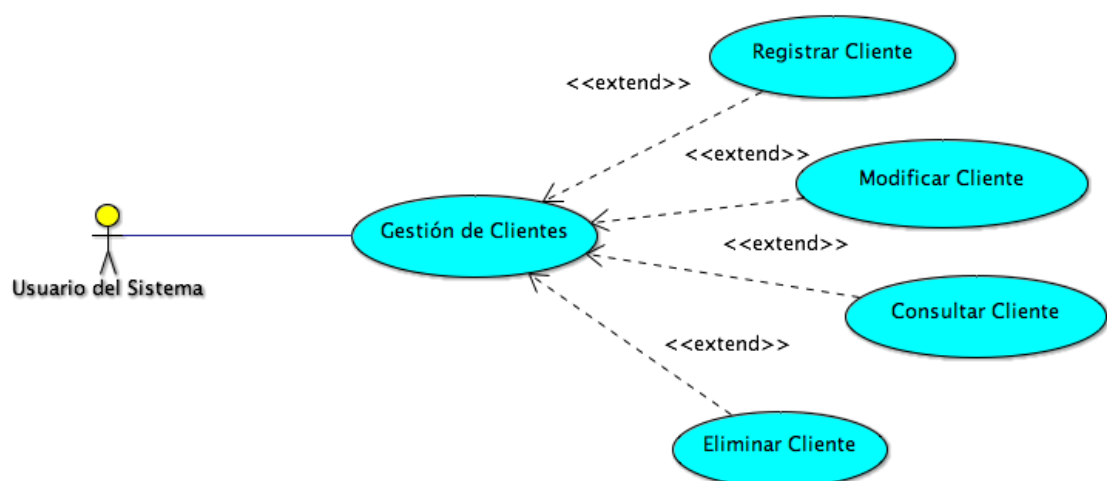


Imagen 14: Caso de Uso de Gestión de Clientes

➤ **Caso de uso de Función de Gestión de Tipos de Persona**

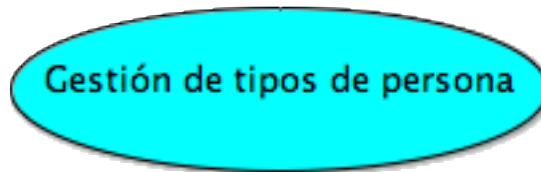


Imagen 15: Caso de Uso de Gestión de Tipos de Persona

Casos de uso: 6	Función de Gestión de tipos de persona
Actor:	Usuarios
Propósito:	Ingreso y mantenimiento de la información de los tipos de persona
Tipo:	Esencial
Referencia:	R.1.1. Registrar tipo de persona. R.1.2. Modificar tipo de persona. R.1.3. Consultar tipo de persona. R.1.4. Eliminar tipo de persona.

Tabla 17: Función de Gestión de tipos de persona

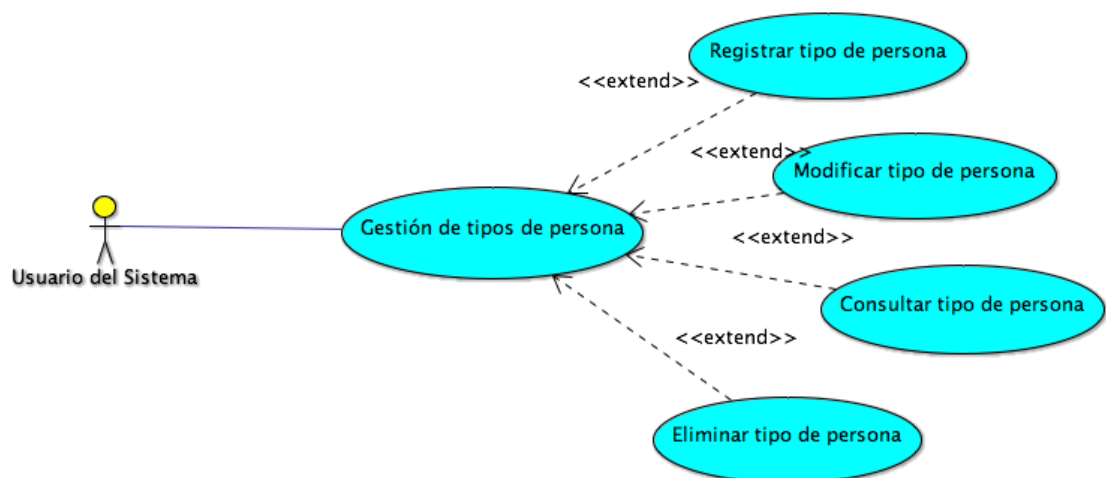


Imagen 16: Caso de Uso de Gestión de Tipos de Persona

➤ **Caso de uso de Función de Gestión de Tipos de Identificación**

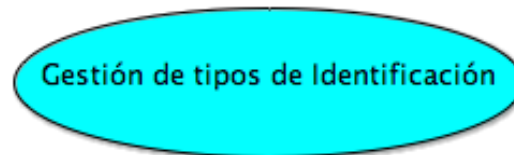


Imagen 17: Caso de Uso de Gestión de Tipos de Identificación

Casos de uso: 7	Función de Gestión de tipos de identificación
Actor:	Usuarios
Propósito:	Ingreso y mantenimiento de tipos de identificación
Tipo:	Esencial
Referencia:	R.1.1. Registrar tipo de identificación. R.1.2. Modificar tipo de identificación. R.1.3. Consultar tipo de identificación. R.1.4. Eliminar tipo de identificación.

Tabla 18: Función de Gestión de tipos de identificación

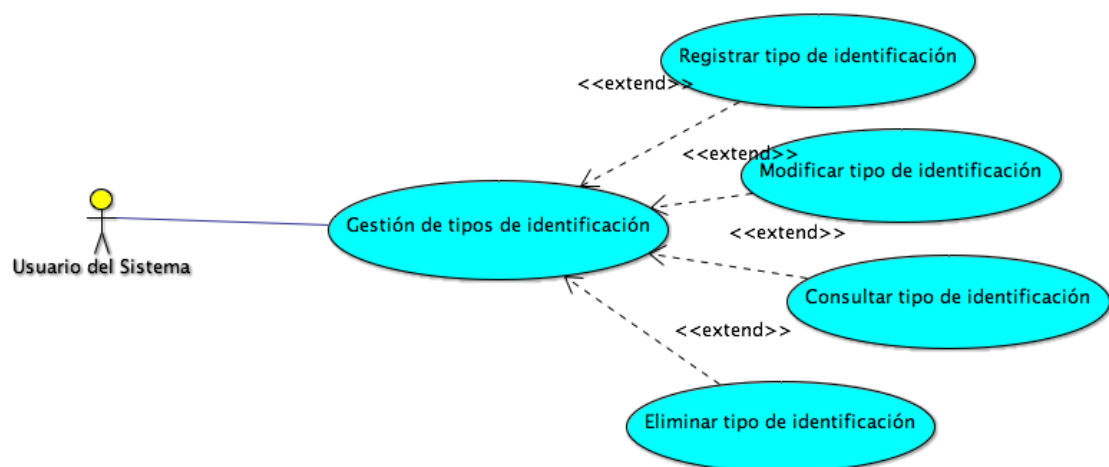


Imagen 18: Caso de Uso de Gestión de Tipos de Identificación

➤ **Caso de uso de Función de Gestión de Tipos de contribuyente**

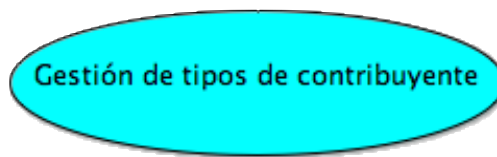


Imagen 19: Caso de Uso de Gestión de Tipos de Contribuyente

Casos de uso: 8	Función de Gestión de tipos de contribuyente
Actor:	Usuarios
Propósito:	Ingreso y mantenimiento de tipos de contribuyente
Tipo:	Esencial
Referencia:	R.1.1. Registrar tipo de contribuyente. R.1.2. Modificar tipo de contribuyente. R.1.3. Consultar tipo de contribuyente. R.1.4. Eliminar tipo de contribuyente.

Tabla 19: Función de Gestión de Tipos de Contribuyente

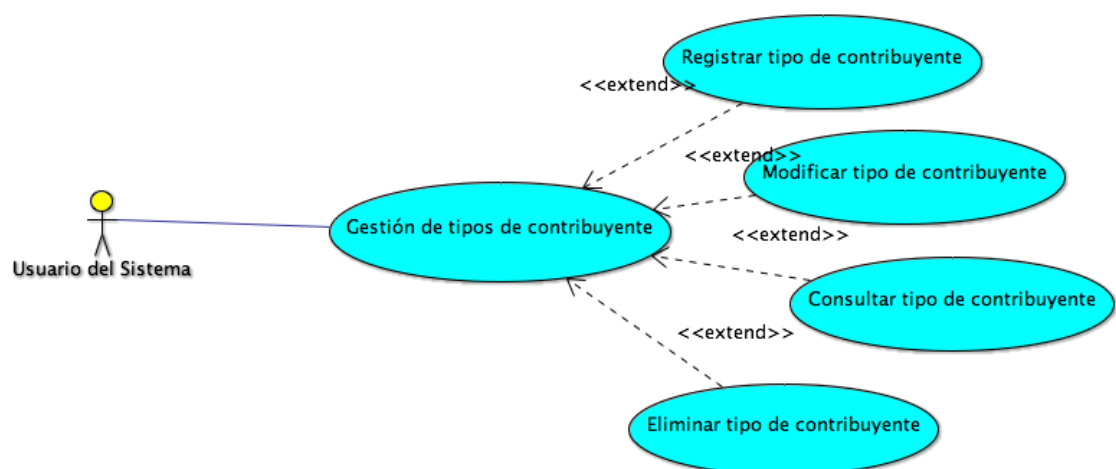


Imagen 20: Caso de Uso de Gestión de Tipos de Contribuyente

➤ **Caso de uso de Función de Gestión de impuestos**

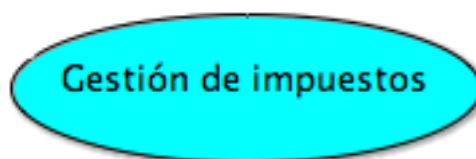


Imagen 21: Caso de Uso de Gestión de Impuestos

Casos de uso: 9	Función de Gestión de impuestos
Actor:	Usuarios
Propósito:	Ingreso y mantenimiento de impuestos
Tipo:	Esencial
Referencia:	R.1.1. Registrar impuesto. R.1.2. Modificar impuesto. R.1.3. Consultar impuesto. R.1.4. Eliminar impuesto.

Tabla 20: Función de Gestión de Impuestos

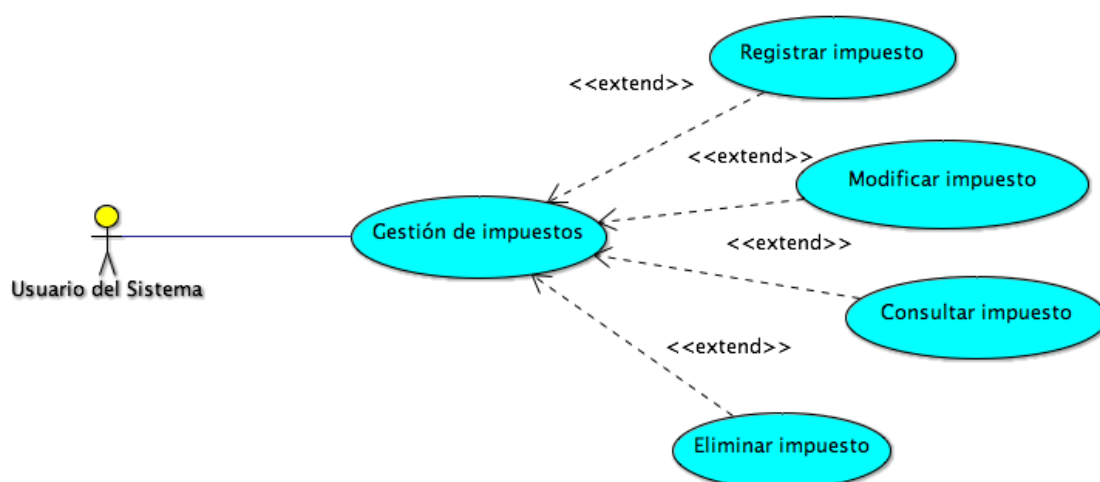


Imagen 22: Caso de Uso de Gestión de Impuestos

➤ Caso de uso de Función de Gestión de códigos de impuestos

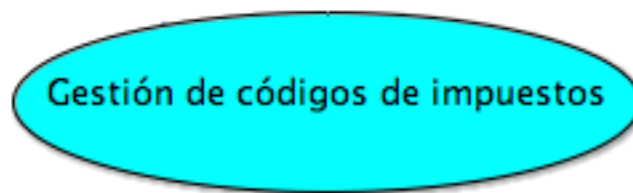


Imagen 23: Caso de Uso de Gestión de Códigos de Impuestos

Casos de uso: 10	Función de Gestión de códigos de impuestos
Actor:	Usuarios
Propósito:	Ingreso y mantenimiento de códigos de impuestos
Tipo:	Esencial
Referencia:	R.1.1. Registrar código de impuesto. R.1.2. Modificar código de impuesto. R.1.3. Consultar código de impuesto. R.1.4. Eliminar código de impuesto.

Tabla 21: Función de Gestión de Códigos de Impuestos

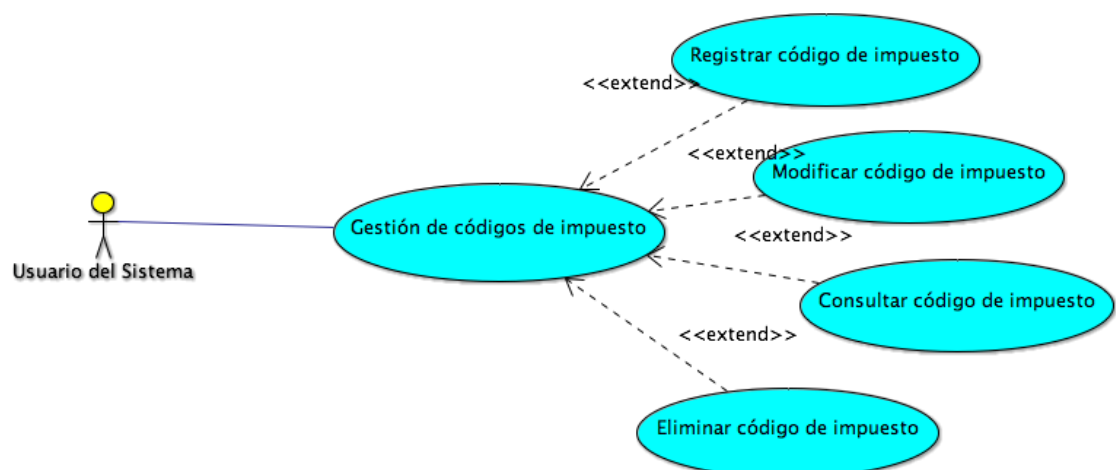


Imagen 24: Caso de Uso de Gestión de Códigos de Impuestos

➤ **Caso de uso de Función de Emisión de Facturación Electrónica**

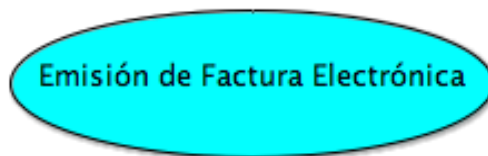


Imagen 25: Caso de Uso de Gestión de Emisión de Factura Electrónica

Casos de uso: 11	Función de Gestión de Factura Electrónica
Actor:	Usuarios.
Propósito:	Ingreso y emisión de facturación electrónica.
Tipo:	Esencial.
Referencia:	R.1.1. Ingresar cliente. R.1.2. Validar información. R.1.3. Ingresar ítems de factura. R.1.4. Emitir Factura Electrónica.

Tabla 22: Función de Gestión de Factura Electrónica

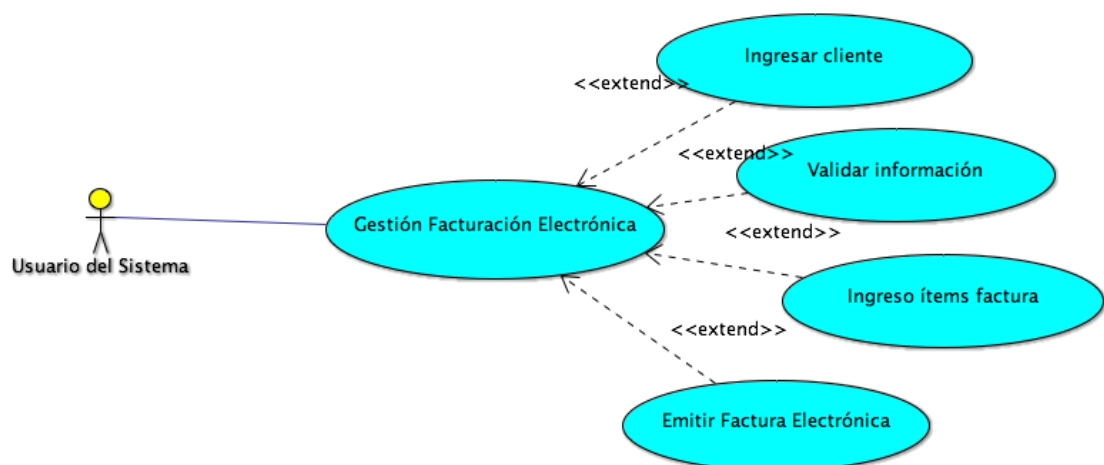


Imagen 26: Caso de Uso de Gestión de Emisión de Factura Electrónica

2.3.2 Diccionario de Datos

➤ **Lista de tablas del modelo entidad – relación**

NOMBRE DE LA TABLA	TIPO DE OBJETO
TCFACTURA	Tabla
TCLIENTES	Tabla
TCODIGOSIMPUESTOS	Tabla
TDFACTURA	Tabla
TDFACTURAIMPUUESTOS	Tabla
TEMPRESAS	Tabla
TIMPUESTOS	Tabla
TITEMS	Tabla
TITEMSIMPUESTOS	Tabla
TPERSONAS	Tabla
TTIPOSCONTRIBUYENTE	Tabla
TTIPOSIDENTIFICACION	Tabla
TTIPOSPERSONA	Tabla
TUSUARIOS	Tabla

Tabla 23: Listado de Tablas del Modelo Entidad - Relación

➤ **Lista de referencias primarias**

REFERENCIA	TABLA	NOMBRE RESTRICCIÓN
IPKCODIMPUESTO	TCODIGOSIMPUESTOS	PK_TCODIGOSIMPUESTOS
IPKTCFACTURA	TCFACTURA	PK_TCFACTURA
IPKTCLIENTES	TCLIENTES	PK_TCLIENTES
IPKTDFACIMPUESTO	TDFACTURAIMPUESTOS	PK_TDFACTURAIMPUESTOS
IPKTDFACTURA	TDFACTURA	PK_TDFACTURA
IPKTEMPRESAS	TEMPRESAS	PK_TEMPRESAS
IPKTITEMS	TITEMS	PK_TITEMS
IPKTIMPUESTOS	TIMPUESTOS	PK_TIMPUESTOS
IPKTITEMIMPUESTO	TITEMSIMPUESTOS	PK_TITEMSIMPUESTOS
IPKTPERSONAS	TPERSONAS	PK_TPERSONAS
IPKTTIPOCONTRI	TTIPOSCONTRIBUYENTE	PK_TTIPOSCONTRIBUYENTE
IPKTTIPOIDENTIFICACION	TTIPOSIDENTIFICACION	PK_TTIPOSIDENTIFICACION
IPKTTIPOSPERSONA	TTIPOSPERSONA	PK_TTIPOSPERSONA
IPKTUSUARIOS	TUSUARIOS	PK_TUSUARIOS

Tabla 24: Listado de Referencias Primarias

➤ **Lista de referencias foráneas**

REFERENCIAS	CODIGO	TABLA PRIMARIA	TABLA SECUNDARIA	COLUMNAS REFERENCIAS
FKCFACTURACLIENTE	FKCFACTURACLIENTE	TCLIENTES	TCFACTURA	CEMPRESA; CCLIENTE
FKCFACTURAUSUARIO	FKCFACTURAUSUARIO	TUSUARIOS	TCFACTURA	CEMPRESA; CUSUARIO
FKCLIENTEMPRESA	FKCLIENTEMPRESA	TEMPRESAS	TCLIENTES	CEMPRESA
FKCLIPERSONA	FKCLIPERSONA	TPERSONAS	TCLIENTES	CPERSONA
FKDETFACTURACAB	FKDETFACTURACAB	TCFACTURA	TDFACTURA	CEMPRESA; CNUMEROFACTURA
FKDFACIMPDFACTURA	FKDFACIMPDFACTURA	TDFACTURA	TDFACTURAIMPUESTOS	CEMPRESA; CNUMEROFACTURA; CITEM
FKDFACIMPTEMPRESA	FKDFACIMPTEMPRESA	TEMPRESAS	TDFACTURAIMPUESTOS	CEMPRESA
FKDFACTURAEMPRESA	FKDFACTURAEMPRESA	TEMPRESAS	TDFACTURA	CEMPRESA
FKEMPRESAPERSONA	FKEMPRESAPERSONA	TPERSONAS	TEMPRESAS	CPERSONA
FKFACTURAEMPRESA	FKFACTURAEMPRESA	TEMPRESAS	TCFACTURA	CEMPRESA
FKITEMEMPRESA	FKITEMEMPRESA	TEMPRESAS	TITEMS	CEMPRESA
FKPERSONAEMPRESA	FKPERSONAEMPRESA	TEMPRESAS	TPERSONAS	CEMPRESA
FKPERSONATIPOCONTRI	FKPERSONATIPOCONTRI	TTIPOSCONTRIBUYENTE	TPERSONAS	CTIPOCONTRIBUYENTE
FKPERTIPOIDENT	FKPERTIPOIDENT	TTIPOSIDENTIFICACION	TPERSONAS	CTIPOIDENTIFICACION
FKPERTIPOPER	FKPERTIPOPER	TTIPOSPERSONA	TPERSONAS	CTIPOPERSONA
FKTDFACIMPTITEMIMP	FKTDFACIMPTITEMIMP	TITEMSIMPUESTOS	TDFACTURAIMPUESTOS	CEMPRESA; CITEM; CIMPUESTO; CCODIGOIMPUESTO
FKTDFACTURATITEM	FKTDFACTURATITEM	TITEMS	TDFACTURA	CEMPRESA; CITEM
FKTIMPTCODIMP	FKTIMPTCODIMP	TCODIGOSIMPUESTOS	TIMPUESTOS	CCODIGOIMPUESTO
FKTITEMIMPTCODIMP	FKTITEMIMPTCODIMP	TCODIGOSIMPUESTOS	TITEMSIMPUESTOS	CCODIGOIMPUESTO
FKTITEMIMPTEMPRESA	FKTITEMIMPTEMPRESA	TEMPRESAS	TITEMSIMPUESTOS	CEMPRESA
FKTITEMIMPTIMPUESTO	FKTITEMIMPTIMPUESTO	TIMPUESTOS	TITEMSIMPUESTOS	CIMPUESTO; CCODIGOIMPUESTO
FKTITEMIMPTITEM	FKTITEMIMPTITEM	TITEMS	TITEMSIMPUESTOS	CEMPRESA; CITEM
FKUSUARIOEMPRESA	FKUSUARIOEMPRESA	TEMPRESAS	TUSUARIOS	CEMPRESA
FKUSUARIOPERSONA	FKUSUARIOPERSONA	TPERSONAS	TUSUARIOS	CPERSONA

Tabla 25: Listado de Referencias Foráneas

➤ **Lista de columnas de la tabla TCFACTURA**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CEMPRESA	int			X	X	X
CNUMEROFACTURA	varchar(20)	20		X		X
CCLIENTE	int				X	X
CUSUARIO	varchar(50)	50			X	X
SECUENCIAFACTURA	int					X
FEMISION	date					X
PUNTOEMISION	varchar(3)	3				X
PUNTOTRABAJO	varchar(3)	3				X
NUMEROAUTORIZACION	varchar(100)	100				X
OBSERVACIONES	varchar(200)	200				
TOTALSINIMPUESTOS	numeric(19,6)	19	6			X
TOTALCONIMPUESTOS	numeric(19,6)	19	6			X
TOTALIMPUESTOS	numeric(19,6)	19	6			X
VALORDESCUENTO	numeric(19,6)	19	6			X
TOTALGENERAL	numeric(19,6)	19	6			X
ESTADO	int					X
ESTADO_SRI	int					X

Tabla 26: Listado de Campos de la tabla TCFACTURA

➤ **Lista de columnas de la tabla TCLIENTES**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CEMPRESA	int			X	X	X
CCLIENTE	int			X		X
CPERSONA	int				X	X
ESTADO	int					X

Tabla 27: Listado de Campos de la tabla TCLIENTES

➤ **Lista de columnas de la tabla TCODIGOSIMPUESTOS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CCODIGOIMPUESTO	int			X		X
DESCRIPCION	varchar(100)	100				

Tabla 28: Listado de Campos de la tabla TCODIGOSIMPUESTOS

➤ **Lista de columnas de la tabla TIMPUESTOS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CIMPUESTO	int			X		X
CCODIGOIMPUESTO	int			X		
DESCRIPCION	varchar(100)	100				
PORCENTAJE	numeric(19, 2)					
ESTADO	int					X

Tabla 29: Listado de Campos de la tabla TIMPUESTOS

➤ **Lista de columnas de la tabla TDFACTURA**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANE	MANDATORIO
CEMPRESA	int			X	X	X
CNUMEROFACTURA	varchar(20)	20		X	X	X
CITEM	int			X	X	X
CANTIDAD	numeric(19,6)	19	6			X
PRECIO	numeric(19,6)	19	6			X
SUBTOTAL	numeric(19,6)	19	6			X
TOTALIMPUESTOS	numeric(19,6)	19	6			X
DESCUENTO	numeric(19,6)	19	6			X

Tabla 30: Listado de Campos de la tabla TDFACTURA

➤ **Lista de columnas de la tabla TDFACTURAIMPUESTOS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANE	MANDATORIO
CEMPRESA	int			X	X	X
CNUMEROFACTURA	varchar(20)	20		X	X	X
CCODIGOIMPUESTO	Int			X	X	X
CIMPUESTO				X	X	X
CITEM	int			X	X	X
PORCENTAJE	numeric(19,6)	19	6			X
VALOR	numeric(19,6)	19	6			X

Tabla 31: Listado de Campos de la tabla TDFACTURAIMPUESTOS

➤ **Lista de columnas de la tabla TEMPRESAS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CEMPRESA	int			X		X
CPERSONA	int				X	X
ESTADO	int					X
NUMEROCONTRIBUYENTE_	int					
LLEVACONTABILIDAD	varchar(1)	1				

Tabla 32: Listado de Campos de la tabla TEMPRESAS

➤ **Lista de columnas de la tabla TITEMS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CEMPRESA	int			X	X	X
CITEM	int			X		X
ESTADO	int					X
DESCRIPCION	varchar(100)	100				
PRECIO	numeric(19,6)	19	6			X

Tabla 33: Listado de Campos de la tabla TITEMS

➤ **Lista de columnas de la tabla TITEMSIMPUESTOS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CEMPRESA	int			X		X
CITEM	int			X		X
CCODIGOIMPUESTO	int			X		X
CIMPUESTO	Varchar(2)			X		X
ESTADO	int					X

Tabla 34: Listado de Campos de la tabla TITEMSIMPUESTOS

➤ **Lista de columnas de la tabla TPERSONAS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CPERSONA	int			X		X
CTIPOPERSONA	varchar(3)	3			X	X
CTIPOIDENTIFICACION	varchar(3)	3			X	X
CTIPOCONTRIBUYENTE	int				X	X
IDENTIFICACION	varchar(13)	13				X
ESTADO	int					X
PRIMERNOMBRE	varchar(50)	50				
SEGUNDONOMBRE	varchar(50)	50				
PRIMERAPELLIDO	varchar(50)	50				
SEGUNDOAPELLIDO	varchar(50)	50				
NOMBRELEGAL	varchar(200)	200				
RAZONSOCIAL	varchar(200)	200				
REPRESENTALTELEGAL	varchar(200)	200				
DIRECCION	varchar(200)	200				
TELEFONOCASA	varchar(20)	20				
TELEFONOCELULAR	varchar(20)	20				
EMAIL	varchar(100)	100				
CEMPRESA	int				X	

Tabla 35: Listado de Campos de la tabla TPERSONAS

➤ **Lista de columnas de la tabla TTIPOSCONTRIBUYENTE**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CTIPOCONTRIBUYENTE	int			X		X
DESCRIPCION	varchar(100)	100				
ESTADO	int					X

Tabla 36: Listado de Campos de la tabla TTIPOSCONTRIBUYENTE

➤ **Lista de columnas de la tabla TTIPOSIDENTIFICACION**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CTIPOIDENTIFICACION	varchar(3)	3		X		X
CODIGOSRI	varchar(2)	2				X
DESCRIPCION	varchar(100)	100				
ESTADO	int					X

Tabla 37: Listado de Campos de la tabla TTIPOSIDENTIFICACION

➤ **Lista de columnas de la tabla TTIPOSPERSONA**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CTIPOPERSONA	varchar(3)	3		X		X
DESCRIPCION	varchar(100)	100				
ESTADO	int					X

Tabla 38: Listado de Campos de la tabla TTIPOSPERSONA

➤ **Lista de columnas de la tabla TUSUARIOS**

CAMPO	TIPO DE DATO	LONGITUD	PRECISION	LLAVE PRIMARIA	LLAVE FORANEA	MANDATORIO
CEMPRESA	int			X	X	X
CUSUARIO	varchar(50)	50		X		X
PASSWORD	varchar(200)	200				X
CPERSONA	int				X	X
ESTADO	Int					X

Tabla 39: Listado de Campos de la tabla TUSUARIOS

2.4 Diseño de Procesos.

Para el desarrollo de software es importante manejar una estructura orientada al diseño de procesos, lo cual nos permite mejorar la productividad, calidad, organización y gestión de tareas durante su ciclo de vida.

2.4.1 Diagrama de Actividades

Para el diseño de procesos utilizaremos diagramas de actividad que nos mostrarán los procesos de la lógica del negocio como un flujo de trabajo. A través de una serie de actividades, podremos ver la interacción de las personas y los componentes del software que realizan estas acciones.

A continuación se describen procesos de varios tipos utilizando diagramas de actividades.

➤ Diagrama de Secuencia Inicio de Sesión

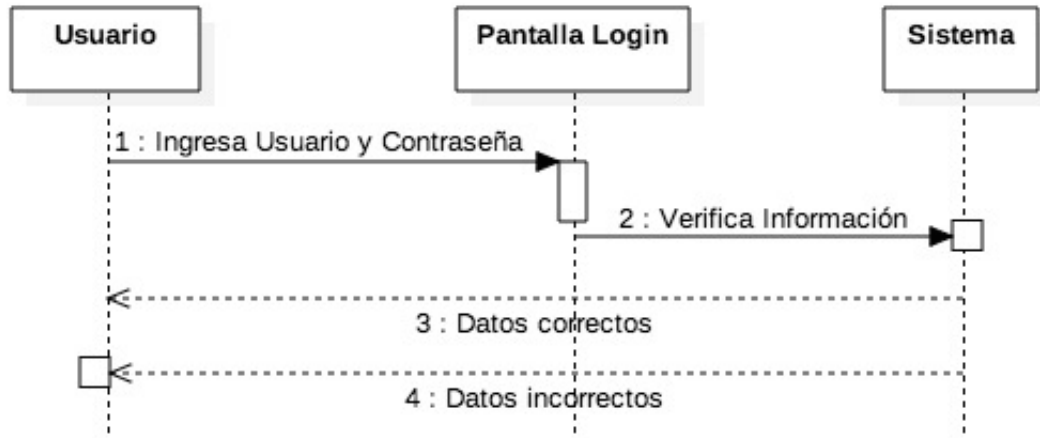


Imagen 28: Diagrama de Secuencia Inicio de Sesión

➤ Diagrama de Secuencia Gestión de Personas

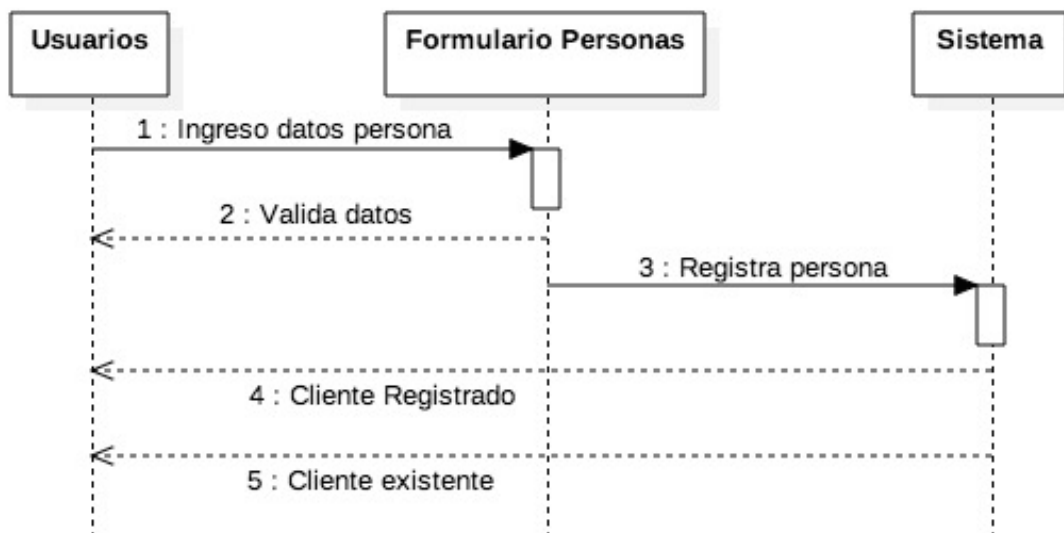


Imagen 29: Diagrama de Secuencia de Gestión de Personas

➤ **Diagrama de Secuencia Gestión de Usuarios**

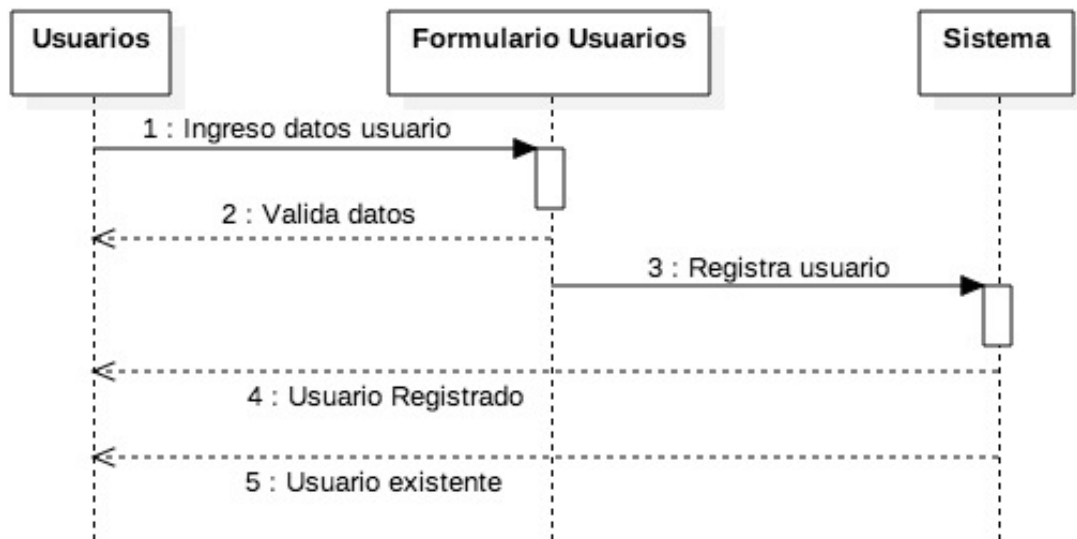


Imagen 30: Diagrama de Secuencia de Gestión de Usuarios

➤ **Diagrama de Secuencia Gestión de Empresas**

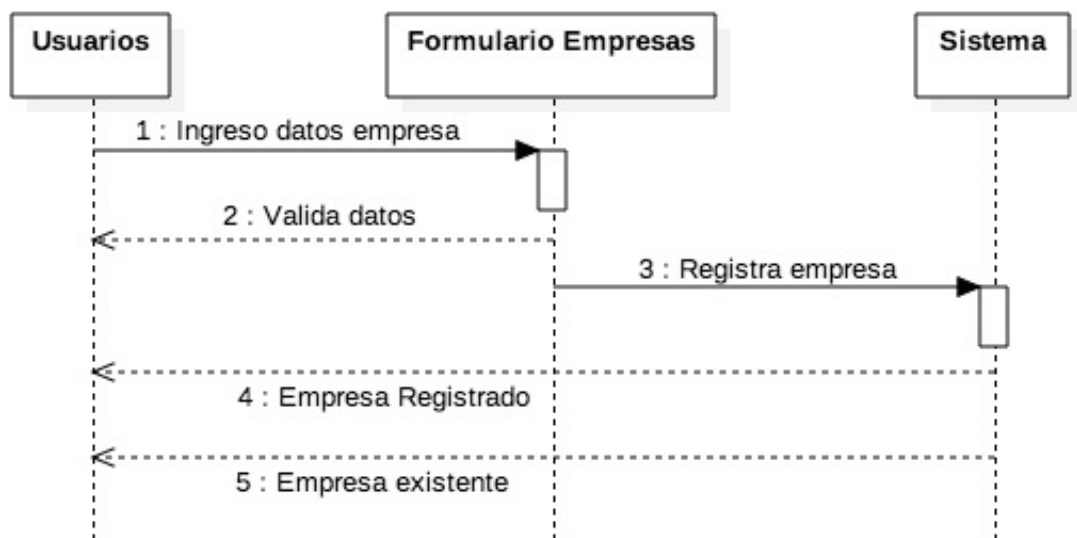


Imagen 31: Diagrama de Secuencia de Gestión de Empresas

➤ **Diagrama de Secuencia Gestión de Ítems**

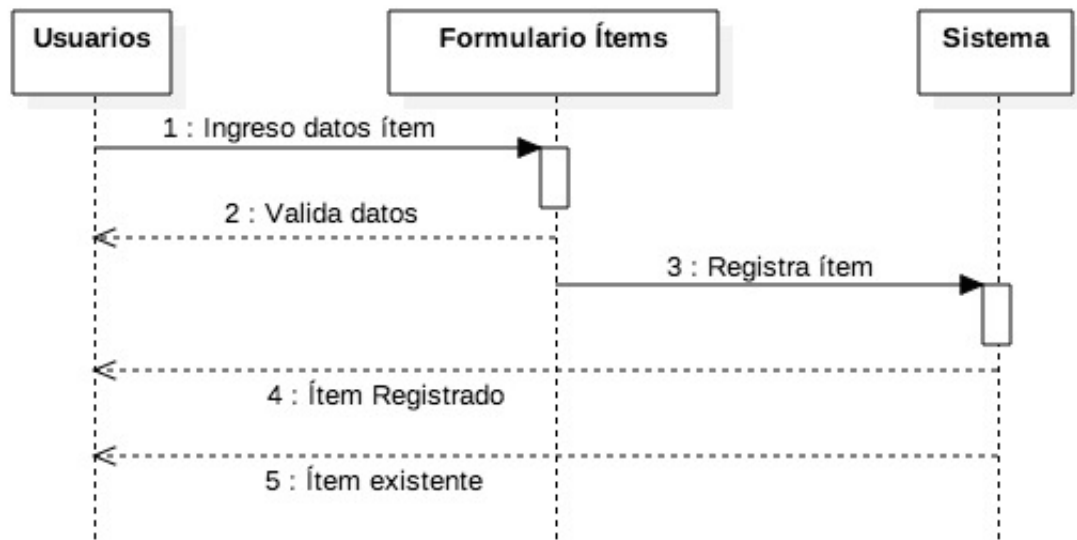


Imagen 32: Diagrama de Secuencia de Gestión de Ítems

➤ **Diagrama de Secuencia Gestión de Clientes**

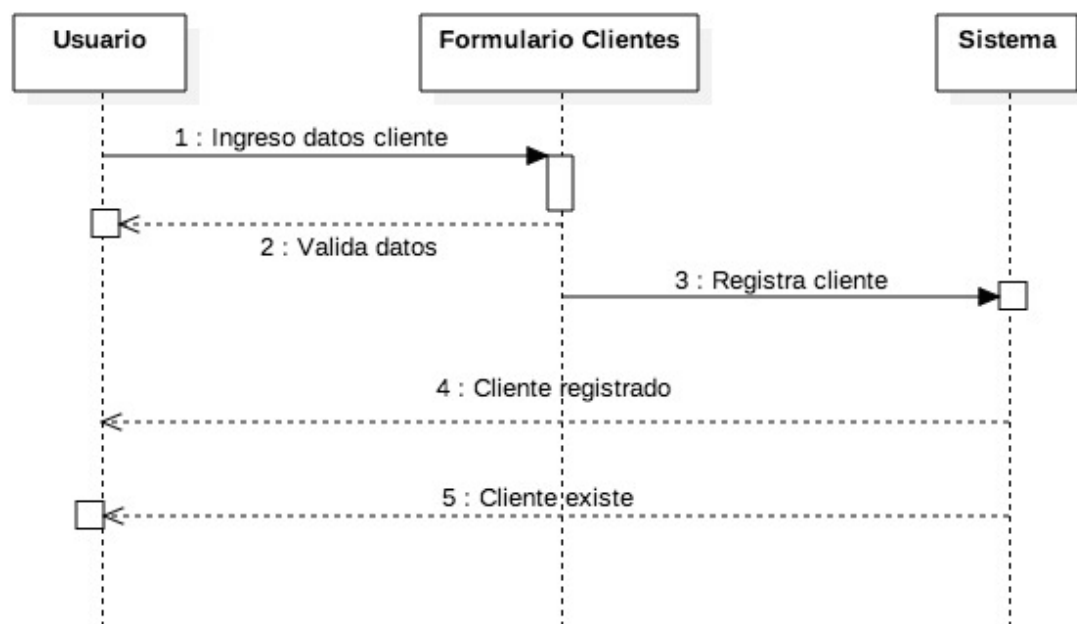


Imagen 33: Diagrama de Secuencia de Gestión de Clientes

Para los siguientes requerimientos, las actividades que intervienen en el flujo de trabajo mantienen el mismo formato que los diagramas anteriores:

- Gestión de Tipos de Persona
- Gestión de Tipos de Identificación
- Gestión de Tipos de Contribuyente
- Gestión de impuestos
- Gestión de Códigos de Impuestos

- **Diagrama Emisión de Factura Electrónica**

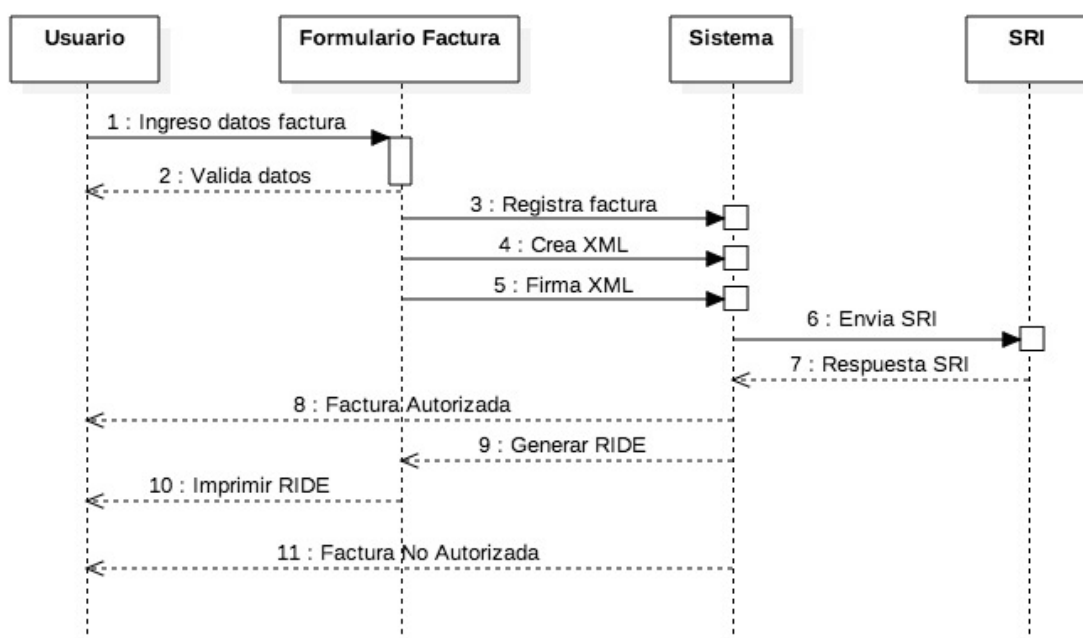


Imagen 34: Diagrama Emisión de Factura Electrónica

2.5 Diseño Arquitectónico.

En un sistema basado en componentes, es necesario implementar una arquitectura que nos permita establecer un marco estructural para identificar los diferentes componentes y la comunicación que existe entre ellos.

El contar con un diseño arquitectónico correcto en el desarrollo de software nos ayuda a negociar y levantar los requerimientos del sistema de una manera correcta, brindando así una mejor comunicación con los clientes, desarrolladores y gestores que están involucrados en el desarrollo de una aplicación.

2.5.1 Diagrama de despliegue

El diagrama de despliegue permite observar los diferentes componentes que intervienen en la implementación del sistema, ya que muestra las relaciones que existen entre los elementos hardware y los procesos que ejecutan los componentes software en el producto final, es decir, modela el tipo de arquitectura de un sistema en tiempo de ejecución.

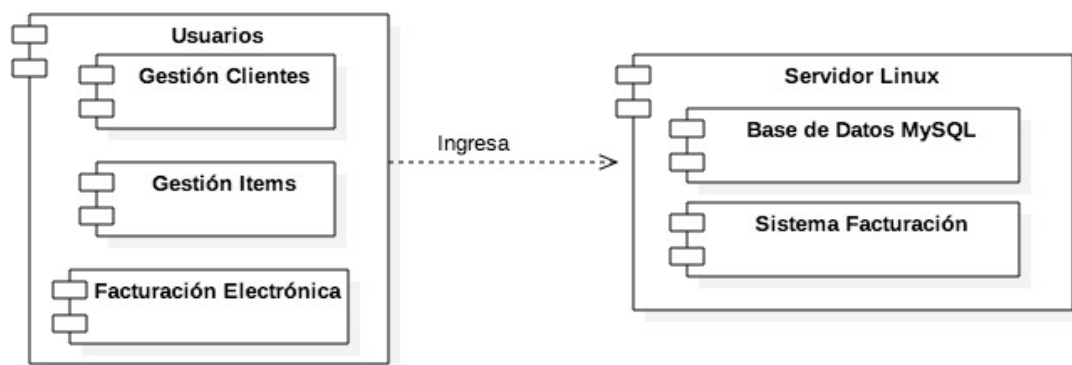


Imagen 35: Diagrama de Despliegue

2.6 Diseño de interfaces.

La interfaz es la parte visible que utiliza el usuario final y permite la interacción con el software y el hardware. Una interfaz debe ser:

- Atractiva
- Intuitiva
- Fácil de usar
- Con respuesta rápida
- Fácil de comprender
- Coherente en toda la pantalla de interfaz

Dichas características aportan una mejora en el diseño del sistema, haciéndolo más atractivo para el cliente, lo cual contribuye a su popularización y facilita su comercialización hacia un mayor número de consumidores.

2.6.1 Describir elementos de la interfaz del sistema

En la siguiente ilustración se muestra la pantalla principal del sistema, en la cual podemos observar el siguiente esquema:

- Posee una cabecera en la parte superior
- Un contenedor principal en el centro.
- Un menú a la izquierda

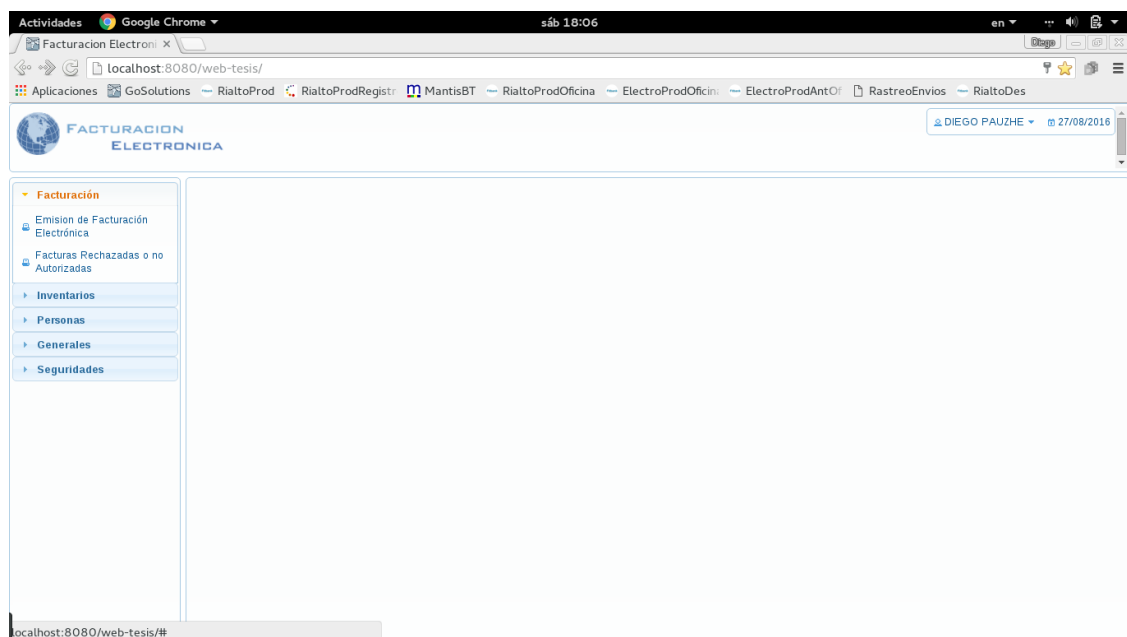


Imagen 36: Pantalla principal del sistema

A continuación se describen los elementos de la interfaz.

➤ Cabecera de la Pantalla



Imagen 37: Pantalla de la cabecera de la pantalla

En la parte izquierda se visualiza el logo del sistema, en la parte derecha están los nombres del usuario que se encuentra en sesión y la fecha actual con la que va a trabajar el sistema.

➤ Menú

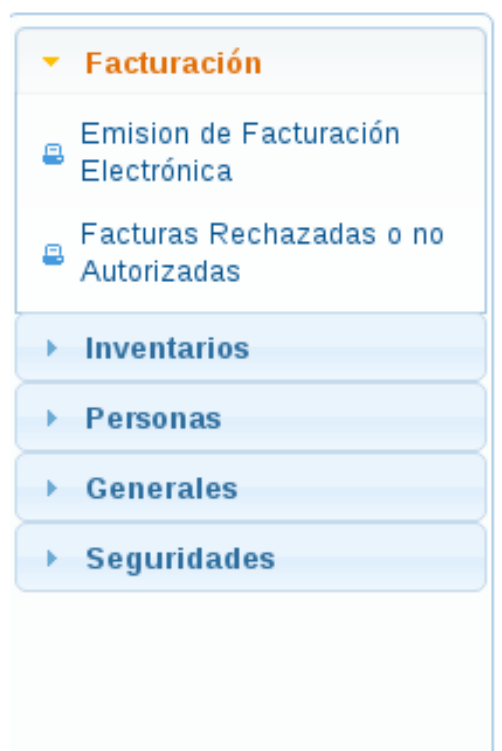


Imagen 38: Menú del Sistema

El menú tiene los accesos a todos los módulos y sus transacciones o ventanas. La navegación es rápida, no se necesita más de dos pantallas para una correcta administración de los mantenimientos desarrollados, lo cual facilita la interacción con el sistema.

➤ Contenedor principal

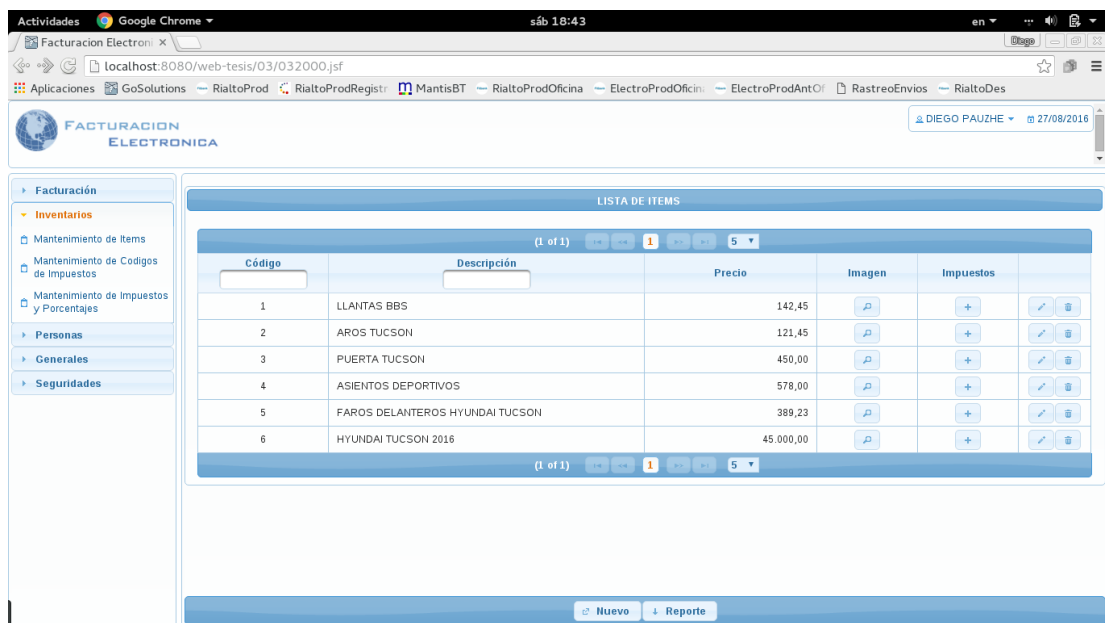


Imagen 39: Cuerpo del sistema, lugar donde se abren todas las pantallas

Está ubicado en el centro del navegador y es en donde se realiza la gestión y administración de todas las transacciones o mantenimientos del sistema. Como ejemplo se describe un listado de ítems que corresponde a la pestaña Inventarios del menú. Las listas y mantenimientos de los demás módulos desarrollados mantienen el mismo esquema y organización descrito.

➤ Lista de Registros

LISTA DE ITEMS					
(1 of 1) << < 1 > >> 5 ▼					
Código	Descripción	Precio	Imagen	Impuestos	
1	LLANTAS BBS	142,45		+	 
2	AROS TUCSON	121,45		+	 
3	PUERTA TUCSON	450,00		+	 
4	ASIENTOS DEPORTIVOS	578,00		+	 
5	FAROS DELANTEROS HYUNDAI TUCSON	389,23		+	 
6	HYUNDAI TUCSON 2016	45.000,00		+	 
(1 of 1) << < 1 > >> 5 ▼					

Imagen 40: Formato de lista de registros

Cada mantenimiento tiene una lista de los registros ingresados en la cual se puede consultar el detalle de cada uno. También se muestran botones para visualizar imágenes, agregar información, editar información, eliminar información, campos de búsqueda, botones de paginación (que son muy útiles cuando se tiene gran cantidad de datos) y un combo box con el número de registros que se desea mostrar.

➤ Botón para mostrar imágenes



Se utiliza este botón cuando un registro almacena una imagen. Despliega la imagen en una ventana independiente.

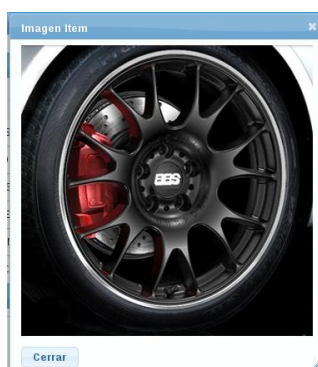


Imagen 41: Pantalla de visualización de imágenes

➤ **Botón para agregar datos**



Esta opción abre una ventana, también conocida como Pop Up (Dialog), que nos muestra un formulario de mantenimiento el cual permite agregar o eliminar datos correspondientes a cada registro. En el ejemplo se tiene más de un impuesto para el ítem seleccionado. El formulario nos permite agregar o eliminar n registros.

Modificación de Códigos

Código: 1 Descripción: LLANTAS BBS

Código Impuesto: Seleccione Impuestos: Seleccione

Agregar Cancelar

Impuestos		Codigo de Impuestos		Porcentaje	
Código	Descripción	Código	Descripción		
3	IVA 14%	2	IVA	14.00	
0001	NO LLEVA ICE	3	ICE	0.00	

Imagen 42: Pantalla de modificación de datos

➤ **Botón para editar datos**



Permite modificar los datos de los registros en una ventana de diálogo, de esta manera no se necesita abandonar la pantalla actual.

(1 of 1) 1 5

Código	Descripción	Precio	Imagen	Impuestos	
1	LLANTAS BBS	142,45		+	
2		121,45		+	
3		450,00		+	
4		578,00		+	
5		389,23		+	
6		45.000,00		+	

Modificación de Ítems

Código: 1 Descripción: LLANTAS BBS

Precio: 142,45

+ Buscar + Subir + Cancelar

Modificar Cancelar

Imagen 43: Dialog para modificación de datos

Campos adicionales de un dialog.

Modificar: Guarda los cambios que se hagan al registro.

Cancelar: Cancela y cierra la venta de modificación.

Buscar: Permite navegar por los archivos de la computadora en busca de la imagen a subir.

Subir: Cuando se tiene seleccionada la imagen, este botón la sube al servidor o base de datos según sea el caso.

Cancelar: Cancela la subida del archivo o imagen.

➤ **Botón para eliminar datos**

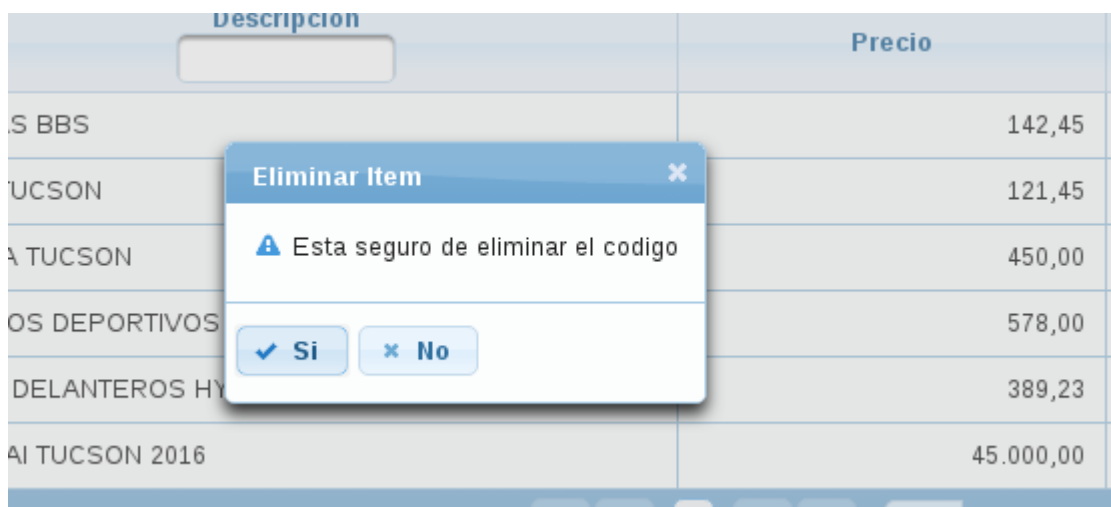


Imagen 44: Dialog para eliminación de registros

Esta opción permite eliminar el registro, el cual abre una ventana de confirmación para asegurarse que se desea eliminar o no. Es necesario aclarar que ningún registro se elimina de la base de datos sino que se aplica eliminación lógica. De esa manera se tiene un historial y mantienen los registros en el sistema.

➤ Campos de búsqueda



Código	Descripción	Precio
1	LLANTAS BBS	142,45

Imagen 45: Filtros de búsqueda

Permite una búsqueda más exacta y rápida, ingresando varios criterios (por ejemplo el código o la descripción del registro), los cuales pueden registrarse total o parcialmente. En base a esta información automáticamente se filtran todos los registros que tengan coincidencia con los datos ingresados.

➤ Botón nuevo y botón Reporte



Imagen 46: Formato de botones

Reporte: Descarga un archivo en formato PDF con los datos que se han configurado en el reporte. Estos reportes son creados con la herramienta iReport 5.6.0.

Nuevo: Abre una nueva pantalla con un formulario nuevo, el cual nos permite ingresar un nuevo registro.

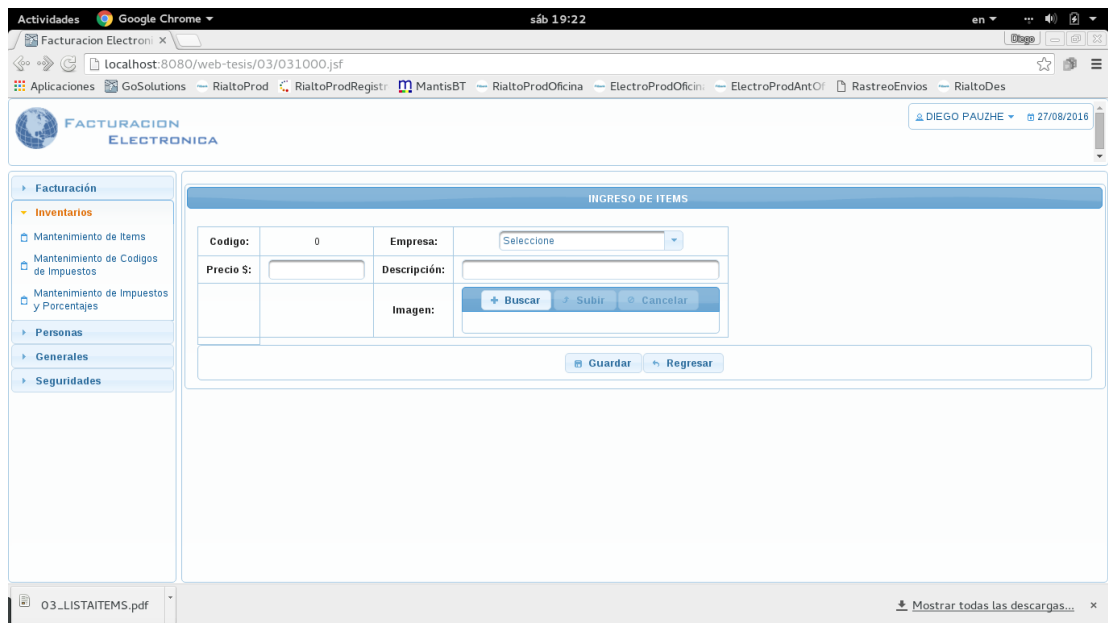


Imagen 47: Formato de pantalla para el mantenimiento de datos

Guardar: Permite grabar el nuevo registro después de haber ingresado los datos correspondientes.

Regresar: Regresa a la pantalla anterior.

CAPÍTULO 3

DESARROLLO DE LA APLICACIÓN

3.1 Introducción.

En este último capítulo se detallan los campos obligatorios o necesarios para la creación del archivo XML basado en la Ficha Técnica publicada en la página del SRI.

Se muestra el proceso que se debe considerar para la aprobación del documento electrónico, describiendo cada campo que es parte del archivo y adjuntando un ejemplo del RIDE que es el resultado final del sistema.

Para todo lo mencionado es necesario realizar diferentes tipos de pruebas del aplicativo para demostrar que su funcionamiento garantice el normal flujo que lleva a la autorización de dicho documento, así como también los posibles errores con sus respectivos mensajes del sistema.

3.2 Descripción de la arquitectura del software

Para el desarrollo de esta aplicación utilizaremos como arquitectura de software el modelo MVC (Modelo - Vista - Controlador), el cual nos permitirá construir la aplicación en tres componentes distribuidos en los siguientes directorios:

- 1) ./tesis_facturacion/negocio-tesis.- Este directorio corresponde al componente "Modelo" el cual almacena los datos y procesos que representarán la información acerca de la lógica del negocio, como por ejemplo: las clases para ingresar una nueva persona, clases para crear un nuevo ítem, clases correspondientes a la emisión de la Facturación Electrónica, etc.

2) ./tesis_facturacion/web-tesis.- Este directorio corresponde a la "Vista" la cual almacena lo referente a la interfaz del usuario final como por ejemplo: páginas de logueo, formularios para ingreso, modificación y consultas de datos, etc.

Para la organización y desarrollo de este componente seguiremos la siguiente estructura para nombrar los formularios:

021000.xhtml

- Los dos primeros números corresponden a los subsistemas:

Subsistema	Descripción
00	Usuarios
01	Generales
02	Personas
03	Inventarios
04	Facturación

Tabla 40: Número de Módulos utilizados en el sistema

- Los cuatro últimos números corresponden al rango para numerar los formularios:

Transacción	Descripción
1000	Mantenimientos
2000	Consultas/Listados
3000	Reportes/Archivos

Tabla 41: Número de Transacciones utilizadas

3) ./tesis_facturacion/persistence-tesis.- Dentro de este directorio tenemos el componente "Controlador" en donde almacenamos las clases generadas por Hibernate, las cuales nos permiten realizar el mapeo de la base de datos con las clases que interactúan con los procesos que realiza el usuario. Este componente enlaza el modelo con la vista.

3.3 Creación de mantenimientos indispensables.

Para el almacenamiento de los datos que corresponden a la representación de la lógica del negocio, nuestro sistema posee los siguientes mantenimientos:

- Mantenimiento de personas
- Mantenimiento de empresas
- Mantenimiento de usuarios
- Mantenimiento de ítems
- Mantenimiento de tipos de persona
- Mantenimiento de tipos de identificación
- Mantenimiento de tipos de contribuyente
- Mantenimiento de clientes
- Mantenimiento de impuestos
- Mantenimiento de códigos de impuestos

3.4 Creación de las clases necesarias para formar el XML de la factura electrónica.

Para el desarrollo de las clases necesarias para construir el XML estándar que cumpla con la estructura establecida por el Servicio de Rentas Internas para la Facturación Electrónica, hemos tomado en consideración la siguiente información, tablas, códigos, estructuras, etc., la cual fue extraída de la Ficha Técnica de Comprobantes Electrónicos, Versión 2.06, vigente desde Junio 2016 proporcionada por el SRI.

3.4.1 Información requerida para la construcción del XML

Para la emisión de la factura electrónica generaremos el comprobante electrónico en formato .XML de acuerdo a la estructura del esquema .XSD proporcionado por el Servicio de Rentas Internas, el cual está disponible en su sitio Web.

➤ **Tabla 1**

La clave de acceso estará compuesta de 49 caracteres numéricos, la cual se genera automáticamente mediante un proceso interno que hemos desarrollado en este proyecto de titulación. Esta clave es única y corresponde al número de autorización del comprobante electrónico, con el cual el SRI responderá de manera satisfactoria o no.

No.	Descripción de campo	Tipo de campo	Formato	Longitud	Requisito	Etiqueta o tag en archivo XML
1	Fecha de Emisión	Numérico	ddmmaaaa	8	Obligatorio	<claveAcceso>
2	Tipo de Comprobante		00	2		
3	Número de RUC		1234567890001	13		
4	Tipo de Ambiente		Tabla 5	1		
5	Serie*		001001	6		
6	Número Secuencial*		000000001	9		
7	Código Numérico		Numérico	8		
8	Tipo de Emisión		Tabla 2	1		
9	Dígito Verificador (módulo 11)		Numérico	1		

Imagen 48: Descripción de caracteres que forman parte de la clave de acceso

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

➤ **Tabla 2**

La tabla muestra el código del tipo de emisión que formará parte de la clave de acceso del comprobante electrónico.

No.	Tipo de Emisión	Código	Requisito
1	Emisión Normal ¹	1	Obligatorio

Imagen 49: Tipo de emisión del comprobante

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

➤ **Tabla 4**

Esta tabla presenta los tipos de comprobantes electrónicos con sus respectivos códigos, los cuáles pueden ser generados por los contribuyentes

No.	Nombre comprobante	Código	Requisito	Etiqueta o tag en archivo XML
1	FACTURA	01	Obligatorio	<codDoc>
2	NOTA DE CRÉDITO	04		
3	NOTA DE DÉBITO	05		
4	GUÍA DE REMISIÓN	06		
5	COMPROBANTE DE RETENCIÓN	07		

Imagen 50: Comprobantes que se puede emitir electrónicamente

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

Para el desarrollo de este trabajo de titulación nos centraremos en el código 01 FACTURA. Este código forma parte de la clave de acceso de cada comprobante que generaremos, la cual es única en los documentos electrónicos.

➤ **Tabla 5**

Se muestra a continuación los tipos de ambiente para la emisión y recepción de los documentos electrónicos. Para las pruebas que se realizarán en este proyecto se usará siempre el código 1 PRUEBAS. Con la herramienta desarrollada se puede configurar el tipo de ambiente desde un archivo properties ubicado en el servidor de aplicaciones.

No.	Tipo de ambiente	Código	Requisito
1	Pruebas	1	Obligatorio
2	Producción	2	

Imagen 51: Tipos de ambientes utilizados para la emisión

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-**

line[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

➤ **Tabla 6**

Esta tabla nos muestra los diferentes tipos de identificación, las cuales se almacenan en la tabla TPERSONA. Este código también forma parte de la clave de acceso del comprobante electrónico.

No.	Tipo de identificación	Código	Requisito
1	RUC	04	Obligatorio
2	CEDULA	05	Obligatorio
3	PASAPORTE	06	Obligatorio
4	VENTA A CONSUMIDOR FINAL*	07	Obligatorio
5	IDENTIFICACION DELEXTERIOR*	08	Obligatorio
6	PLACA	09	Obligatorio

Imagen 52: Tipos de identificación que se puede utilizar para el cliente

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

➤ **Tabla 9**

Esta tabla muestra los datos o códigos que forman parte de la Clave de Acceso que se generará automáticamente mediante un proceso interno de la aplicación al momento de emitir la factura. A continuación se describe cada campo de esta tabla:

No.	Descripción de campo	Tipo de campo	Formato	Longitud	Requisito	Etiqueta o tag en archivo XML
1	Fecha de Emisión	Numérico	ddmmaaaa	8	Obligatorio	<claveAcceso>
2	Tipo de Comprobante		00	2		
3	Número de RUC		1234567890001	13		
4	Tipo de Ambiente		Tabla 5	1		
5	Serie*		001001	6		
6	Número Secuencial*		000000001	9		
7	Código Numérico		Numérico	8		
8	Tipo de Emisión		Tabla 2	1		
9	Dígito Verificador (módulo 11)		Numérico	1		

Imagen 53: Campos que forman parte de la clave de acceso

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

- 1 Fecha de Emisión. Es la fecha en la que se envía el documento electrónico y debe estar en el formato (ddmmaaaa).
- 2 Tipo de Comprobante. Es el código del tipo de comprobante para la emisión de la factura electrónica. Corresponde al código 01 FACTURA (Tabla 2).
- 3 Número de RUC. Es el número de RUC del contribuyente agente quien emite el comprobante electrónico.
- 4 Tipo de Ambiente. Es el tipo de ambiente en el que se está facturando y puede ser de Pruebas o de Producción (Tabla 5).
- 5 Serie. Es la unión del Establecimiento y Punto de Emisión de dónde se está facturando, por ejemplo 001-001.
- 6 Número Secuencial. Es la secuencia de la factura que consta de 9 dígitos. Mantiene el mismo formato que las anteriores facturas físicas, como por ejemplo 000000001.
- 7 Código Numérico. Es un código cualquiera de 8 dígitos que no tiene referencia a ningún código en particular.
- 8 Tipo de Emisión. Es el tipo de emisión. En este caso es Emisión Normal (Tabla 2)
- 9 Dígito Verificador. “El dígito verificador será aplicado sobre toda la clave de acceso (48 dígitos) y deberá ser incorporado por el contribuyente a través del método denominado Módulo 11, con un factor de chequeo ponderado (2). Este mecanismo de detección de errores, será verificado al momento de la recepción del comprobante. Cuando el resultado del dígito verificador obtenido sea igual a once (11), el dígito verificador será el cero (0) y cuando el resultado del dígito verificador obtenido sea igual a diez 10, el dígito verificador será el uno (1). El código numérico constituye un mecanismo para brindar seguridad al emisor en cada comprobante emitido. El algoritmo numérico para conformar este código es potestad absoluta del contribuyente emisor.”. (Electrónicos, 2015)

Ejemplo de verificación utilizando algoritmo de módulo 11:

Cadena de verificación: 41261533

Pasos 1 y 2

4	1	2	6	1	5	3	3	-	?
x3	x2	x7	x6	x5	x4	x3	x2		
=12	=2	=14	=36	=5	=20	=9	=6		

Paso 3 12 +2 +14 +36 +5 +20 +9 +6 = **104**

Paso 4 **104** mod **11** = 5 (ya que 104 = 11 x 9 + 5)

Paso 5 11 - 5 = 6 **Resultado = 6**

Imagen 54: Formato de calcular el dígito verificador mediante el Módulo 11

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

➤ **Tabla 10**

Esta tabla contiene los códigos de los impuestos.

Impuesto	Código
IVA	2
ICE	3
IRBPNR	5

Imagen 55: Códigos de Impuestos

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

➤ **Tabla 11**

Esta tabla muestra el código del porcentaje de la tarifa del IVA que se va a utilizar en la facturación electrónica.

Porcentaje de IVA	Código
0%	0
12%	2
14%	3
No Objeto de Impuesto	6
Exento de IVA	7

Imagen 56: Porcentajes de Impuestos

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

3.5 Proceso de firma del comprobante electrónica

Es obligatorio firmar cada archivo XML en todos los comprobantes electrónicos generados, los cuales deben cumplir con el estándar de documentos XML: XadES_BES. Esto quiere decir que cada fichero XML debe contener dentro de su estructura la firma electrónica y una vez aprobado por el SRI, se convertirá en un comprobante electrónico válido.

Descripción	Especificación	Documentación técnica relacionada
Estándar de firma	XadES_BES	http://uri.etsi.org/01903/v1.3.2/ts_101903v010302p.pdf
Versión del esquema	1.3.2	http://uri.etsi.org/01903/v1.3.2#
Codificación	UTF-8	
Tipo de firma	ENVELOPED	http://www.w3.org/2000/09/xmldsig#enveloped-signature

Imagen 57: Formato de firma electrónica

Fuente: **Ficha Técnica de Comprobantes Electrónicos Esquema Off-line**

[Disponible en: <http://www.sri.gob.ec/web/guest/10116>]

A continuación se describen las normas técnicas que constituyen el estándar de documentos XML: La estructura del formato básico de firma electrónica avanzada acorde con la presente política se adecua a las especificaciones definidas en XADES_BES, que incluyen los campos que se describen en el esquema 1.3.2 del cuadro anterior.

La firma electrónica se considera un nodo más a añadir en el documento .XML. El nivel de seguridad en la firma electrónica está ejecutado sobre tres partes de la trama de datos:

- Todos los elementos o nodos que conforman el comprobante electrónico.
- Los elementos de firma ubicados en el contenedor “SignedProperties”.
- El certificado digital con el que se ha firmado incluido en el elemento “KeyInfo”.

Es necesario utilizar el elemento ds:KeyInfo, conteniendo al menos el certificado firmante codificado en base64. Además dicha información precisa ser firmada con objeto de evitar la posibilidad de sustitución del certificado.

Cada comprobante deberá incorporar la firma electrónica en formato XADES-Bes, misma que se puede realizar con librerías destinadas para el efecto. El SRI utilizó el siguiente set de librerías para incorporar y validar la firma de cada comprobante:

- MITyCLibXADES
- MITyCLibTSA
- MITyCLibAPI
- MITyCLibOCSP
- MITyCLibTrust

A continuación se detallan las especificaciones técnicas relacionadas al algoritmo de encriptación:

Algoritmo de Firmado: RSA-SHA1

Longitud de clave: 2048 bits. Recomendación técnica basada en documento: http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1_revised2_Mar08-2007.pdf

Archivo de Intercambio de Información: PKCS12 (extensión .p12). Este archivo deberá ser proporcionado ya sea de manera directa (a través de API's de acceso al token USB), o de manera indirecta a través de la extracción del mismo y posterior instalación en una carpeta específica desde la cual el software proporcionado por el SRI lo utilizará para firmar los comprobantes. Para la aplicación de nuestra tesis hemos adquirido el certificado .p12, el cual fue comprado ingresando a: <https://www.eci.bce.ec/web/guest/solicitud-de-certificado-requisitos>. El archivo será instalado en nuestro servidor en la ruta /Tesis/sri/jks/ y la aplicación obtendrá la información requerida del certificado instalado en la ruta especificada.

A continuación se presentan los servicios propuestos en la Web para la autorización en línea de Documentos Electrónicos

Mediante el uso protocolos SSL de seguridad se garantiza la comunicación y estandarización segura entre los servicios expuestos por el Servicio de Rentas Internas.

Dirección Electrónica que recepta los XML que nosotros enviamos desde nuestra aplicación.

<https://celcer.sri.gob.ec/comprobantes-electronicos-ws/RecepcionComprobantesOffline?wsdl>

Dirección Electrónica en la cual se puede verificar el estado de los comprobantes electrónicos enviados.

<https://celcer.sri.gob.ec/comprobantes-electronicos-ws/AutorizacionComprobantesOffline?wsdl>

3.5.1 Descripción de clases que firma el XML.

A continuación se describen los procesos que se ejecutan luego de haber detallado los datos indispensables para la creación del xml y las clases necesarias para dicha generación.

En primer lugar se utilizan las clases que se generan de forma automática desde el archivo .xsd que tomamos de la página del SRI: <http://www.sri.gob.ec/web/guest/10116>

Esquemas XSD y XML Versión 1.0.0 y 2.0.0. Al descargar este comprimido .zip tenemos todos los .xsd de los documentos electrónicos. Con el comando xjc (Linux) se generan las siguientes clases:

- Destino.java
- DetalleImpuestos.java
- Factura.java
- Impuesto.java
- InfoTributaria.java
- ObjectFactory.java
- ObligadoContabilidad.java
- Pagos.java
- Reembolsos.java
- Rubro.java

Las cuales llamamos desde nuestra clase CreateVoucherXML para crear y llenar el xml.

Luego de crear el xml, procedemos a firmar en el formato XADES-Bes utilizando las siguientes clases:

- SignVoucher
- SignatureUtil
- PassStoreKS

Creación de las clases que envían y consultan el estado del comprobante electrónico.

Internamente luego de firmar el documento, se llama a la clase que envía y consulta su validez en el servicio web del SRI. Se utilizan las siguientes clases.

- SendVoucherXML
- SendVoucherThread

Generar el RIDE para la impresión del comprobante.

Finalmente se crea el RIDE para el envío al cliente, por medio de las clases:

- DataFiller
- ReportManager
- ReportType

3.6 Estructura XML de una Factura Electrónica

ETIQUETAS O TAGS	CARACTER	TIPO DE CAMPO	LONGITUD/FORMAT O
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>	Obligatorio	-	-
<factura id="comprobante" version="1.1.0">	Obligatorio	-	-
<infoTributaria>	Obligatorio	-	-
<ambiente>1</ambiente>	Obligatorio conforme Tabla 5	Numerico	1
<tipoEmision>1</tipoEmision>	Obligatorio conforme Tabla 2	Numérico	1
<razonSocial>TESIS FACTURACION</razonSocial>	Obligatorio	Alfanumérico	Max 300
<nombreComercial>TESIS FACTURACION</nombreComercial>	Obligatorio cuando corresponda	Alfanumérico	Max 300
<ruc>0105247076001</ruc>	Obligatorio	Numérico	13

<claveAcceso>2706201601 0105247076001 10010020000000330000003311</claveAcceso>	Obligatorio	Numérico	49
<codDoc>01</codDoc>	Obligatorio conforme Tabla 4	Numérico	2
<estab>001</estab>	Obligatorio	Numérico	3
<ptoEmi>002</ptoEmi>	Obligatorio	Numérico	3
<secuencial>000000033</secuencial>	Obligatorio	Numérico	9
<dirMatriz>SIDCAY</dirMatriz>	Obligatorio	Alfanumérico	Max 300
</infoTributaria>	Obligatorio	-	-
<infoFactura>	Obligatorio	-	-
<fechaEmision>27/06/2016</fechaEmision>	Obligatorio	Fecha	dd/mm/yyyy
<dirEstablecimiento>EUCALIPTOS</dirEstablecimiento>	Obligatorio cuando corresponda	Alfanumérico	Max 300
<contribuyenteEspecial>345</contribuyenteEspecial>	Obligatorio cuando corresponda	Alfanumérico	Max 13
<obligadoContabilidad>SI</obligadoContabilidad>	Obligatorio cuando corresponda	Texto	2

<tipoIdentificacionComprador>05</tipoIdentificacionComprador>	Obligatorio	Numérico	2
<razonSocialComprador>PAUL ALEJANDRO TORRES RIVERA</razonSocialComprador>	Obligatorio	Alfanumérico	Max 300
<identificacionComprador>0104034590</identificacionComprador>	Obligatorio	Alfanumérico	Max 20
<totalSinImpuestos>522.25</totalSinImpuestos>	Obligatorio	Numérico	Max 14
<totalDescuento>70.00</totalDescuento>	Obligatorio	Numérico	Max 14
<totalConImpuestos>	Obligatorio	-	-
<totalImpuesto>	Obligatorio	-	-
<codigo>2</codigo>	Obligatorio conforme Tabla 10	Numérico	1
<codigoPorcentaje>3</codigoPorcentaje>	Obligatorio conforme Tabla 11	Numérico	Min 1 Max 4
<baseImponible>522.25</baseImponible>	Obligatorio	Numérico	Max 14
<valor>73.12</valor>	Obligatorio	Numérico	Max 14
</totalImpuesto>	Obligatorio	Numérico	Max 14
</totalConImpuestos>	Obligatorio	-	-
<propina>0</propina>	Obligatorio	Numérico	Max 14
<importeTotal>595.37</importeTotal>	Obligatorio	Numérico	Max 14
</infoFactura>	Obligatorio	-	-

<detalles>	Obligatorio	-	-
<detalle>	Obligatorio	-	-
<codigoPrincipal>1</codigoPrincipal>	Obligatorio	Alfanumérico	Max 25
<codigoAuxiliar>1</codigoAuxiliar>	Obligatorio cuando corresponda	Alfanumérico	Max 25
<descripcion>LLANTAS BBS</descripcion>	Obligatorio	Alfanumérico	Max 300
<cantidad>1.00</cantidad>	Obligatorio	Numérico	Max 14
<precioUnitario>142.25</precioUnitario>	Obligatorio	Numérico	Max 14
<descuento>25.00</descuento>	Obligatorio	Numérico	Max 14
<precioTotalSinImpuesto>117.25</precioTotalSinImpuesto>	Obligatorio	Numérico	Max 14
<impuestos>	Obligatorio	-	-
<impuesto>	Obligatorio	-	-
<codigo>2</codigo>	Obligatorio conforme Tabla 10	Numérico	1
<codigoPorcentaje>3</codigoPorcentaje>	Obligatorio conforme Tabla 11	Numérico	Min 1 Max 4
<tarifa>14.00</tarifa>	Obligatorio conforme	Numérico	Max 14

	Tabla 11		
<baseImponible>117.25</baseImponible>	Obligatorio	Numérico	Max 14
<valor>16.42</valor>	Obligatorio	Numérico	Max 14
</impuesto>	Obligatorio	-	-
</impuestos>	Obligatorio	-	-
</detalle>	Obligatorio	-	-
<detalle>	Obligatorio	-	-
<codigoPrincipal>3</codigoPrincipal>	Obligatorio	Alfanumérico	Max 25
<codigoAuxiliar>3</codigoAuxiliar>	Obligatorio	Alfanumérico	Max 25
<descripcion>PUERTA TUCSON</descripcion>	Obligatorio	Alfanumérico	Max 300
<cantidad>1.00</cantidad>	Obligatorio	Numérico	Max 14
<precioUnitario>450.00</precioUnitario>	Obligatorio	Numérico	Max 14
<descuento>45.00</descuento>	Obligatorio	Numérico	Max 14
<precioTotalSinImpuesto>405.00</precioTotalSinImpuesto>	Obligatorio	Numérico	Max 14
<impuestos>	Obligatorio	-	-
<impuesto>	Obligatorio	-	-
<codigo>2</codigo>	Obligatorio conforme Tabla 10	Numérico	1

<codigoPorcentaje>3</codigoPorcentaje>	Obligatorio conforme Tabla 11	Numérico	Min 1 Max 4
<tarifa>14.00</tarifa>	Obligatorio conforme Tabla 11	Numérico	Max 14
<baseImponible>405.00</baseImponible>	Obligatorio	Numérico	Max 14
<valor>56.70</valor>	Obligatorio	Numérico	Max 14
</impuesto>	Obligatorio	-	-
</impuestos>	Obligatorio	-	-
</detalle>	Obligatorio	-	-
</detalles>	Obligatorio	-	-
</factura>	Obligatorio	-	-

Tabla 42: Estructura XML de una Factura Electrónica

3.7 RIDE de la Factura Electrónica



**FACTURACION
ELECTRONICA**

R.U.C.: 0105247076001

F A C T U R A

No. 001-001-000000007

NÚMERO DE AUTORIZACIÓN:
0608201601010524707600110010010000000070000000718

FECHA Y HORA DE 2016-08-06

AMBIENTE: PRUEBAS

EMISIÓN: NORMAL

CLAVE DE ACCESO



0608201601010524707600110010010000000070000000718

FACTURACION ELECTRONICA

FACTURACION ELECTRONICA

Dirección SIDCAY

Contribuyente Especial Nro 000

OBLIGADO A LLEVAR CONTABILIDAD NO

Razon Social / Nombres y Apellidos: JUAN LUIS GUERRA MOLINA Identificación 0103558086

Fecha Emisión (dd/mm/yyyy): 06/08/2016 Guía

Cod. Principal	Cod. Auxiliar	Cantidad	Descripción	P. Unitario	Descuento	Precio Total
2	2	4.00	AROS TUCSON	121.45	22.00	463.80
SUBTOTAL CON						463.80
SUBTOTAL						0.00
SUBTOTAL No objeto de						0.00
SUBTOTAL Exento de						0.00
SUBTOTAL SIN						463.80
TOTAL Descuento						22.00
ICE						0
IVA						64.93
VALOR TOTAL						528.73

Información Adicional

DIRECCION CLIENTE: TOTORACocha

TELEFONO CLIENTE: 0988554422

E-MAIL: nandop845@gmail.com

Imagen 58: RIDE o PDF de la Factura Electrónica

3.8 Pruebas del aplicativo

En las siguientes pruebas se demuestra la correcta operación del sistema, mostrando su funcionalidad para cumplir con el objetivo final, que es la autorización de la Factura Electrónica. Se inicia levantando la Base de Datos con todas las tablas que se crearon para la comunicación con el sistema, luego se inicia el servidor Jboss (versión WildFly 10.0) que se instaló dentro del IDE Eclipse. De esta manera se puede manipular el sistema desde el navegador web Google Chrome, siendo recomendable utilizar la última versión para un funcionamiento más óptimo.

3.8.1 Pruebas de funcionamiento basado en casos de uso

La primera pantalla que se presenta es la de login, donde se podrá ingresar el código de usuario y contraseña.

Si el usuario digita mal sus datos, el sistema mostrará los respectivos mensajes de error y no permitirá el ingreso.

Si los datos son correctos se muestra la pantalla principal en donde se puede apreciar:

- El logo del aplicativo en la parte superior izquierda.
- El nombre del usuario que se encuentra en la sesión y la fecha actual del sistema, en la parte superior derecha.
- A la izquierda se encuentra el menú de todos los módulos que se crearon para el proyecto con sus respectivos mantenimientos.
- A la derecha se mostrarán las opciones seleccionadas.

La estructura de los mantenimientos es estándar en todas las pantallas.

Al desplegar cualquiera de los módulos se muestra una consulta de los datos que se han ingresado. En todas las pantallas tendremos botones de modificación y eliminación de los registros. En la parte inferior se tiene un botón para la creación de nuevos registros y, si es el caso, un reporte de los datos que se muestra en pantalla.

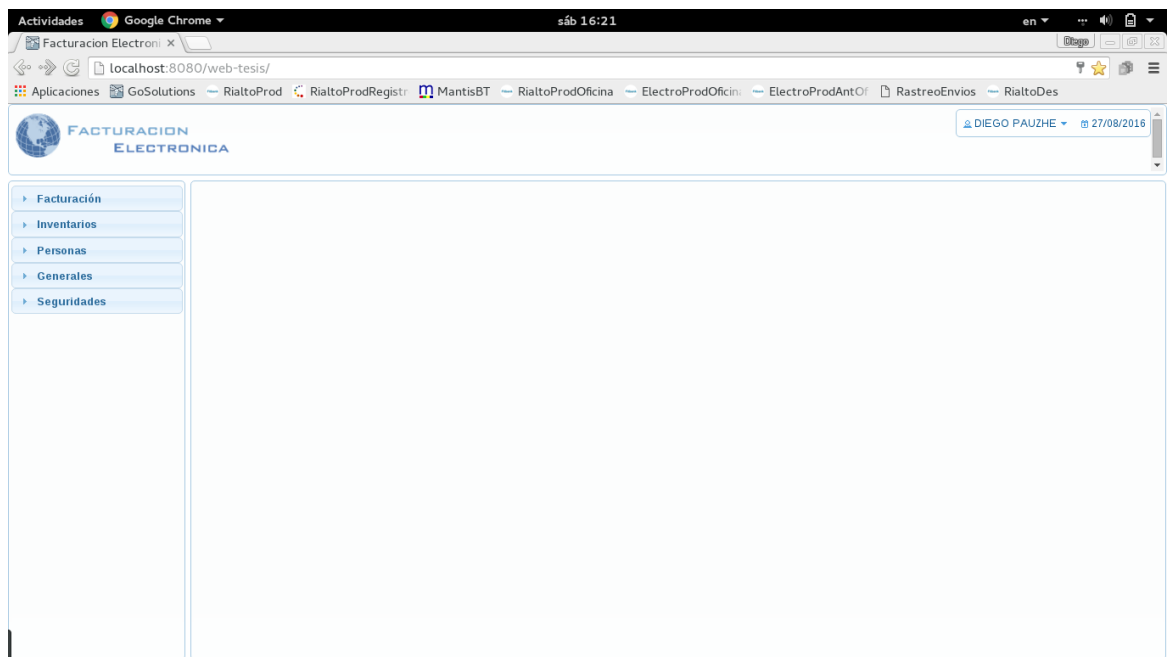


Imagen 59: Pantalla principal del sistema

Dichos botones pueden variar dependiendo de las necesidades de los datos, un claro ejemplo son los ítems, los mismos que tienen botones adicionales para la visualización de imágenes y la asignación de impuestos. Estos botones abrirán una ventana (Dialog) que permitirá al usuario modificar o agregar información sin salir de la interfaz.

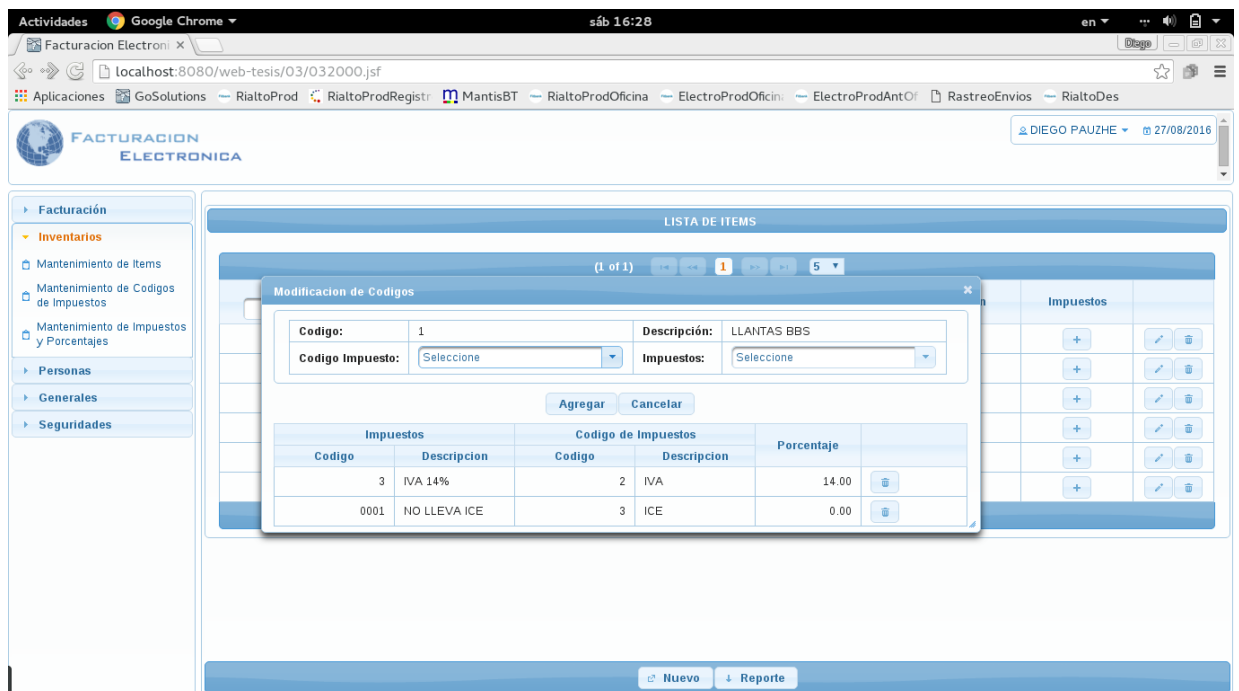


Imagen 60: Elementos que forman parte del Sistema

La función principal del aplicativo es la Factura, en la cual se puede ingresar clientes ya guardados o crear uno nuevo sin tener que abandonar la ventana, para que sea más ágil y rápido. Con respecto al detalle, se pueden agregar ítems (creados previamente) y realizar los cálculos de una manera rápida, fácil y eficiente. Al grabar se hace el proceso de envío y consulta del documento electrónico en tiempo real y finalmente permite al usuario visualizar el RIDE (PDF) del documento autorizado por el SRI.

The screenshot shows a web application titled 'FACTURACION ELECTRONICA'. The interface is divided into a left sidebar with navigation options and a main content area. The main content area is split into two sections: 'DATOS CLIENTE' and 'DETALLE FACTURA'.

Navigation Sidebar:

- Facturación
 - Emission de Facturación Electrónica
 - Facturas Rechazadas o no Autorizadas
- Inventarios
- Personas
- Generales
- Seguridades

DATOS CLIENTE Section:

Cliente:	0302168349	Nombres:	DIEGO FERNANDO PAUZHE HUGO		Dirección:	SIDCAY
Teléfono:	4058252	Celular:	0988632243		Mail:	nandop845@gmail.com
Usuario Facturador:	DPAUZHE	Secuencia:	9		Punto Emisión:	001
Punto Trabajo:	001	Número Documento:	001-001-000000009			
Observaciones:	NUEVAS LLANTAS					

DETALLE FACTURA Section:

Item: Cantidad: Descuento:

Codigo	Descripcion	Cantidad	Precio	Descuento	Valor Impuestos	Subtotal
2	AROS TUCSON	1	121.45	2.00	16.72	119.45
					Subtotal:	119.45
					IVA:	16.72
					ICE:	0
					Descuento:	2.00
					Total:	136.17

Footer Buttons:

Imagen 61: Formato de la pantalla principal de la Factura Electrónica

Al grabar datos en las diferentes pantallas se manejan notificaciones estándar, de esta manera se sabe si se grabó con éxito o si existió algún error, mismo que se detalla.

Transacción guardada con éxito

Cliente:	0105247076	Nombres:	DIEGO ALFREDO ENDERICA ORTEGA		
Teléfono:	4058252	Celular:	0988632243	Mail:	diegoende@hotmail.com
Usuario Facturador:	DPAUZHE	Secuencia:	10	Punto Emisión:	001
Punto Trabajo:	001	Número Documento:	001-001-000000010	Fecha:	27/08/2016
Observaciones:	FACTURA PRUEBA				

DETALLE FACTURA

Item	Cantidad	Descuento	+ Agregar
Seleccione			

Codigo	Descripcion	Cantidad	Precio	Descuento	Valor Impuestos	Subtotal
1	LLANTAS BBS	1	142.45	22.00	16.86	120.45
4	ASIENTOS DEPORTIVOS	2	578.00	28.00	157.92	1128.00
					Subtotal:	1248.45
					IVA:	174.78
					ICE:	0
					Descuento:	50.00
					Total:	1423.23

Guardar PDF + Cliente Nuevo Regresar

Imagen 62: Formato de mensajes de notificaciones

En el caso de que se desee volver a grabar, se realizarán validaciones internas en las transacciones, de manera que no permita el registro de datos duplicados.

Mensaje: No hay datos para guardar, recargue el formulario

Cliente:	0302168349	Nombres:	DIEGO FERNANDO PAUZHE HUGO		
Teléfono:	4058252	Celular:	0988632243	Mail:	nandop845@gmail.com
Usuario Facturador:	DPAUZHE	Secuencia:	9	Punto Emisión:	001
Punto Trabajo:	001	Número Documento:	001-001-000000009	Fecha:	27/08/2016
Observaciones:	NUEVAS LLANTAS				

DETALLE FACTURA

Item	Cantidad	Descuento	+ Agregar
Seleccione			

Codigo	Descripcion	Cantidad	Precio	Descuento	Valor Impuestos	Subtotal
2	AROS TUCSON	1	121.45	2.00	16.72	119.45
					Subtotal:	119.45
					IVA:	16.72
					ICE:	0
					Descuento:	2.00
					Total:	136.17

Guardar PDF + Cliente Nuevo Regresar

Imagen 63: Formato de mensajes de notificación de errores

3.8.2 Pruebas de integración

Durante el desarrollo de este proyecto se observa que la integración entre módulos es una característica fundamental, lo cual brinda una mayor eficiencia y una reducción en el tiempo de ejecución de tareas del usuario, así como del proceso de facturación.

3.8.3 Pruebas de rendimiento (tiempos de consulta, histórico)

A lo largo del desarrollo se realizaron pruebas del rendimiento del aplicativo, los resultados han sido muy favorables, demostrando estabilidad durante todos los procesos.

Los tiempos de respuesta son muy cortos en consultas y grabaciones, los accesos a la base se han reducido para optimizar tiempos, los cuales han sido inferiores a 1 segundo.

El proceso que consume mayor tiempo es el envío del XML al SRI, el cual tiene variaciones que dependen de la congestión que se produce en el *WEB SERVICE* de esta entidad y tiene un tiempo que oscila entre 3 a 8 segundos.

El proceso de autorización ha llegado a dilatarse hasta 15 segundos, los cuales no son perceptibles para el usuario final gracias a un método interno que maneja dicho envío. Este proceso es realizado mediante un Thread (hilo), que se encarga de consultar internamente hasta que obtenga una respuesta positiva del SRI, permitiendo que el aplicativo pueda seguir usándose sin quedarse en un estado de espera.

Un documento podría no ser autorizado por razones que escapen de nuestro alcance como por problemas de conectividad (Internet), falla en el servicio del SRI, caducidad del certificado electrónico, etc.



Imagen 64: Mensaje de indisponibilidad del SRI

Fuente: Web Service del SRI [Disponible en: <https://celcer.sri.gob.ec/comprobantes-electronicos-ws/RecepcionComprobantesOffline?wsdl>]

Para estos casos, se tiene un proceso adicional que nos permitirá el reenvío o consulta de los documentos que no finalizaron su transaccionalidad con éxito.

CONCLUSIONES

A lo largo del desarrollo de la aplicación se demostró la utilidad y ventajas de las herramientas implementadas con las cuales se facilita el desarrollo de sistemas confiables, rápidos y amigables para el usuario, características que son muy necesarias en los sistemas actuales.

Las herramientas Hibernate, Maven y JSF facilitan el desarrollo y la interacción del sistema con la base de datos, haciendo de este un aplicativo que puede llegar a manipular gran cantidad de información de manera rápida y eficiente.

Un servidor JBoss fue de mucha importancia para tener alojado el sistema y acceder a él de forma simple y sin problemas.

Primefaces nos ofrece características que nos ayudan a mejorar la interfaz de usuario, haciéndola más agradable, amigable e intuitiva. Esto permite aumentar la productividad del sistema, facilitando su utilización y aprendizaje.

Con las pruebas realizadas se verificó el correcto funcionamiento del sistema, obteniendo de esta manera un resultado final satisfactorio. Dichas pruebas se generaron con datos reales lo cual demostró la eficiencia del sistema en un entorno de producción.

Como conclusión final podemos mencionar que los objetivos planteados para este proyecto fueron alcanzados satisfactoriamente, obteniendo los resultados esperados y demostrando de esta manera que el uso de un IDE junto con las herramientas propuestas permitieron desarrollar de manera óptima un sistema eficiente, seguro y confiable.

RECOMENDACIONES

Es recomendable utilizar un administrador de proyectos que nos ayude a organizar, integrar y gestionar el desarrollo de una aplicación. Para este proyecto hemos utilizado Maven como una opción viable y de distribución gratuita.

Se debe considerar utilizar un sistema de control de versiones para el desarrollo de software. Esto permitirá al equipo de desarrollo gestionar de una mejor manera proyectos de cualquier tipo y tamaño, llevando así un historial de los cambios que se han realizado en el proceso de desarrollo.

Es aconsejable utilizar una herramienta de mapeo objeto - relacional que facilite la interacción con la base de datos de nuestro proyecto, simplifique su gestión y permita adaptarse a cualquier DBMS. Hibernate es una de las opciones más viables y difundidas en el entorno Java.

ANEXOS

Anexo 1: Integración de las herramientas de desarrollo

Entre algunos de los problemas que existen en la implementación de aplicaciones con Java podemos citar:

- La integración de módulos y proyectos es compleja y consume gran cantidad de tiempo y esfuerzo
- En modelos orientados a objetos, la amplia reutilización de clases externas y sus dependencias, introduce mayor complejidad a la programación
- La gestión del proyecto se vuelve complicada debido a que cada miembro del equipo de desarrollo debe tener instalado el código fuente en cada uno de los equipos en los que se programa, lo cual causa que el código pueda estar desactualizado, haciendo tediosa la integración de los módulos del proyecto e incrementando los tiempos estimados para el desarrollo.
- Al interactuar con la base de datos, generalmente creamos clases para realizar un insert o un update, lo que conlleva a reescribir código en lugar de reutilizarlo. Al tener clases que funcionan de manera exclusiva en ciertas bases de datos, es necesario crear versiones distintas para cada base. Para solventar algunos de estos problemas, dentro del presente proyecto se trabajará con un ambiente integrado de desarrollo en Java, para lo cual se describirá el proceso de instalación de cada una de las herramientas consideradas.

Anexo 2: Instalación del servidor de aplicaciones WildFly

Para este proyecto utilizaremos como servidor local para aplicaciones Java la versión de WildFly que se detalla a continuación:

Versión:	10.0.0.Final
Fecha lanzamiento:	2016-01-29
Descripción:	Java EE7 Full & Web Distribution
Licencia:	LGPL (Lesser General Public License – Licencia Publica General Menor)

WildFly es un servidor de aplicaciones de código abierto y su versión más reciente es la 10.0.0. Se elige este servidor de aplicaciones por ser rápida, ligera y potente.

La siguiente tabla muestra las tecnologías disponibles en la versión 10.0.0 de Wildfly.

Java EE 7 Plataforma Tecnológica	Java EE 7 Perfil Completo	Java EE 7 Perfil Web	WildFly 10 Perfil Completo	WildFly 10 Perfil Web
JSR-356: API Java para Web Socket	X	X	X	X
JSR-353: API de Java para procesamiento de JSON	X	X	X	X
JSR-340: Java Servlet 3.1	X	X	X	X
JSR-344: JavaServer Faces 2.2	X	X	X	X
JSR-341: Expression Language 3.0	X	X	X	X
JSR-245: JavaServer Pages 2.3	X	X	X	X

JSR-52: Estándar para la biblioteca de etiquetas JavaServer Pages (JSTL) 1.2	X	X	X	X
JSR-352: Lote de aplicaciones para la plataforma Java 1.0	X	--	X	--
JSR-236: Utilidades de concurrencia de Java EE 1.0	X	X	X	X
JSR-346: Contextos e inyección de dependencia para Java 1.1	X	X	X	X
JSR-330: Inyección de dependencia para Java 1.0	X	X	X	X
JSR-349: Validación de Beans 1.1	X	X	X	X
JSR-345: Enterprise JavaBeans 3.2	X CMP 2.0 Opcional	X (Lite)	X CMP 2.0 No Disponible	X (Lite)
JSR-318: Interceptores 1.2	X	X	X	X
JSR-322: Arquitectura de conectores Java EE 1.7	X	--	X	X
JSR-338: Persistencia de Java 2.1	X	X	X	X
JSR-250: Anotaciones comunes para la plataforma Java 1.2	X	X	X	X
JSR-343: API de servicios de mensajes Java 2.0	X	--	X	--
JSR-907: API de transacciones Java 1.2	X	X	X	X

JSR-919: JavaMail 1.5	X	--	X	X
JSR-339: API de Java para servicios Web RESTFul 2.0	X	X	X	X
JSR-109: Implementar Servicios Web Empresariales 1.3	X	--	X	--
JSR-224: API de Java para servicios web basados en XML 2.2	X	X	X	X
JSR-181: Servicios Web de Metadatos para la plataforma Java	X	--	X	--
JSR-101: API Java para RPC basada en XML 1.1	Opcional	--	--	--
JSR-67: API Java para mensajería XML 1.3	X	--	X	--
JSR-93: API Java para registros XML	Opcional	--	--	--
JSR-196: Interfaz de proveedor de servicio de autenticación de Java para contenedores 1.1	X	--	X	--
JSR-115: Autorización Java para contrato de contenedores 1.5	X	--	X	--
JSR-88: Implementación de aplicaciones Java EE 1.2	Opcional	--	--	--
JSR-77: Gestión de J2EE 1.1	X		X	
JSR-45: Soporte de depuración para otros lenguajes 1.0	X	X	X	X

Tabla 43: Tecnologías disponibles en la versión 10.0.0 de Wildfly.

Para la instalación se requiere la última actualización disponible de Java SE 8 o posterior. Seguimos los siguientes pasos:

1. Descargamos la versión descrita de la página oficial:
<http://www.wildfly.org/downloads/>
2. Extraemos la descarga en el directorio en donde deseemos almacenar.
3. Se debe explorar la estructura de directorios del servidor de aplicaciones, los archivos de configuración, los archivos logs, etc.

Estructura de directorios Wildfly 10.

DIRECTORIO	DESCRIPCIÓN
appclient	Incluye los archivos de configuración, el contenido de implementación y las áreas de escritura utilizadas por el contenedor del cliente de la aplicación de gestión de la instalación.
bin	Este directorio es en donde se ponen en marcha los scripts de configuración y varias utilidades de línea de comando como Vault, complemento de usuario y informe de diagnóstico de Java disponible para entornos Unix y Windows
bin/client	Contiene un jar cliente para ser usado por clientes no expertos.
docs/schema	Contiene los archivos de definición de esquema XML
docs/examples/configs	Incluye ejemplos de los archivos de configuración para casos específicos
domain	Este directorio incluye los archivos de configuración, contenido de implementación, y

	las áreas de escritura utilizados por el modo de dominio que se ejecutan en procesos de esta instalación.
modules	En este directorio se almacenan los diferentes módulos utilizados en el servidor (Wildfly 10 se basa en una arquitectura modular de carga de clases)
standalone	Guarda los archivos de configuración, contenido de implementación y las áreas de escritura utilizados en el servidor independiente de esta instalación.
welcome-content	Guarda la página de bienvenida por defecto.

Tabla 44: Estructura de directorios Wildfly 10.

Una vez que se ha instalado la aplicación, para iniciar el servidor de aplicaciones Wildfly 10, ingresamos desde la consola al directorio donde descomprimos y ejecutamos el siguiente comando:

```
/Users/diegoenderica/Desktop/tesis/wildfly-10.0.0.Final/standalone.sh
```

El servidor iniciará mostrando información similar a la siguiente:

```
=====

JBoss Bootstrap Environment

JBOSS_HOME: /Users/diegoenderica/jboss

JAVA: /Library/Java/JavaVirtualMachines/jdk1.8.0_73.jdk/Contents/Home/bin/java

JAVA_OPTS: -server -Xms64m -Xmx512m -XX:MetaspaceSize=96M -
XX:MaxMetaspaceSize=256m -Djava.net.preferIPv4Stack=true -
Djboss.modules.system.pkgs=org.jboss.byteman -Djava.awt.headless=true

=====
```

11:46:11,161 INFO [org.jboss.modules] (main) JBoss Modules version 1.5.1.Final

11:46:11,331 INFO [org.jboss.msc] (main) JBoss MSC versión 1.2.6.Final

11:46:11,391 INFO [org.jboss.as] (MSC service thread 1-6) WFLYSRV0049: WildFly Full 10.0.0.Final (WildFly Core 2.0.10.Final) starting

....

11:46:14,300 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Full 10.0.0.Final (WildFly Core 2.0.10.Final) started in 1909ms - Started 267 of 553 services (371 services are lazy, passive or on-demand)

Una vez que el servidor de aplicaciones esté levantado correctamente podemos visualizar la página de bienvenida desde la siguiente url en nuestro navegador: <http://localhost:8080>



Imagen 65: Panta Bienvenida WildFly

Anexo 3: Instalación del administrador de proyectos Maven

Para la construcción de nuestra aplicación utilizaremos la versión 3.3.9 de Maven. Esta herramienta nos permitirá gestionar de una manera adecuada este proyecto.

Requisitos del sistema

Java Development Kit (JDK)	Maven 3.3 requiere JDK 1.7 o superior para ejecutarse (aunque todavía le permite construir para JDK 1.3 y posteriores)
Memoria	No hay requisito mínimo
Disco	Aproximadamente 10 MB es necesario para la propia instalación de Maven. Además de eso, el espacio de disco adicional se utilizará para su repositorio local de Maven. El tamaño de su repositorio local variará dependiendo del uso pero se espera que al menos sea de 500 MB.
Sistema operativo	No existe un requisito mínimo. Programa de inicio se incluyen como shell scripts y archivos por lotes de Windows.

Tabla 45: Requisitos para instalación Maven 3.3.9

Pasos para la instalación de Maven 3.3.9:

- Descargar la última versión de Maven 3.3.9 de la página oficial :
<https://maven.apache.org/download.cgi>
- Descomprimos el archivo que hemos descargado “apache-maven-3.3.9-bin.zip” en el directorio /usr/local/maven_3_3_9

- Configuramos nuestro archivo `/etc/profile` agregando a la variable de entorno `PATH` el directorio `bin` de maven ubicado en `/usr/local/maven_3_3_9`, a continuación se muestra como está configurado el archivo `profile`.

```
=====
JAVA_HOME=/Library/Java/JavaVirtualMachines/jdk1.8.0_92.jdk/Contents/Home
PATH=$JAVA_HOME/bin:$PATH
MAVEN_HOME=/usr/local/maven_3_3_9
PATH=$MAVEN_HOME/bin:$PATH
export JAVA_HOME MAVEN_HOME PATH
=====
```

- Confirmamos que la instalación de maven esté correcta ejecutando el comando `"mvn -version"`, el resultado debe ser algo similar a:

```
=====
Apache Maven 3.3.9 (bb52d8502b132ec0a5a3f4c09453c07478323dc5; 2015-11-10T11:41:47-05:00)
Maven home: /usr/local/maven_3_3_9
Java version: 1.8.0_92, vendor: Oracle Corporation
Java home: /Library/Java/JavaVirtualMachines/jdk1.8.0_92.jdk/Contents/Home/jre
Default locale: es_ES, platform encoding: UTF-8
OS name: "mac os x", version: "10.10.5", arch: "x86_64", family: "mac"
=====
```

Anexo 4: Instalación de Hibernate.

La versión de Hibernate que usaremos para la creación y administración de persistencias en este proyecto es la 4.3.1.Final.

Pasos para la instalación de Hibernate 4.3.1.Final.

- Descargar la última versión de Hibernate de la página oficial:

<http://hibernate.org/orm/downloads/>

- Descomprimir el fichero hibernate-release-4.3.1.Final.zip en la ubicación que desee:

En el directorio que hemos descargado dentro del fichero /lib se encuentran 4 carpetas que contienen las librerías java con el código de Hibernate.

lib/required. Contiene los jars que siempre deben usarse en Hibernate, es decir siempre debemos incluir estos jars en todos nuestros proyectos de Hibernate.

lib/jpa. las librerías necesarias para usar JPA con Hibernate.

lib/optional. Contiene las librerías que añaden nuevas funcionalidades a Hibernate, como poder usar el Pool de conexiones C3PO o el sistema de cache EhCache, etc.

lib/envers. Contiene librerías que permiten realizar auditorías sobre los datos que se persisten.

- Copiar el fichero hibernate-entitymanager-4.3.1.Final.jar de la carpeta lib/jpa en la carpeta lib de nuestro proyecto java.
- Finalmente se puede indicar que queremos usar esas librerías desde nuestro IDE de preferencia, para este proyecto hemos usado Eclipse en su versión Mars 2

Instalación Opcional de Hibernate versión 4.3.1.Final

Se puede instalar y configurar Hibernate directamente desde el IDE Eclipse Mars 2, para esto podemos descargar la última versión de su página oficial:

<http://www.eclipse.org/downloads/packages/release/Mars/2>

Versión: Eclipse Mars.2 Release (4.5.2)

En el menú Help tenemos la opción Eclipse Marketplace, en la cual descargamos el plugin JBoss Tools 4.3.1.Final.

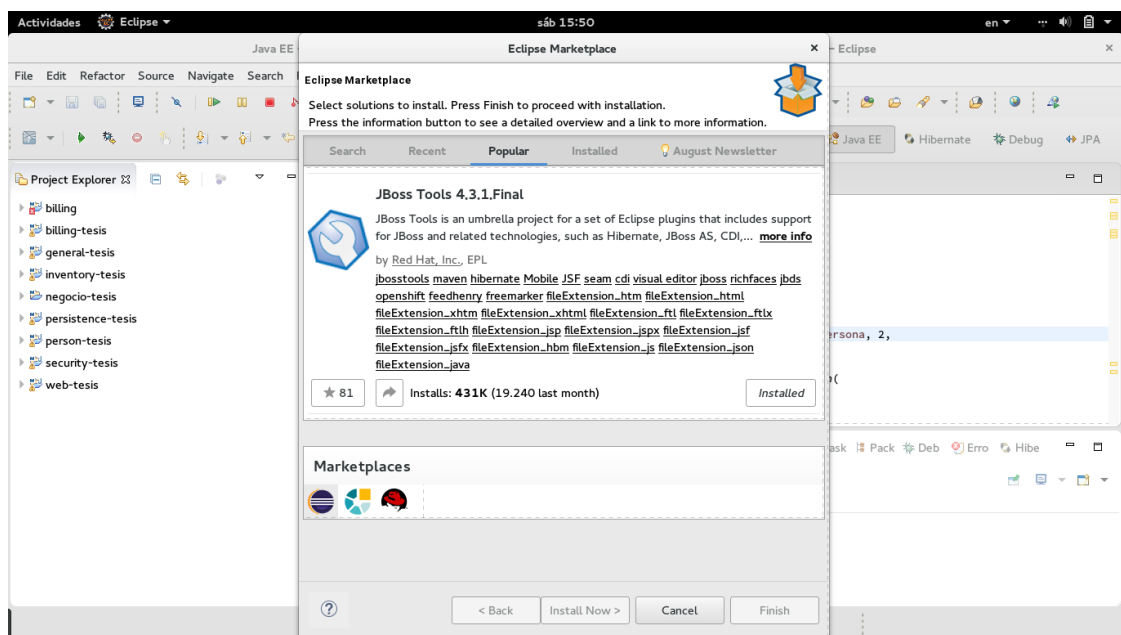


Imagen 66: Eclipse Marketplace - Eclipse Mars.2 Release (4.5.2)

Este plugin contiene 3 herramientas importantes que se describen a continuación:

Servidor Jboss: Es el servidor de aplicaciones web que se encarga del alojamiento de la aplicación.

Maven: Se encarga de la administración y gestión de las dependencias del proyecto.

Hibernate: Nos servirá para la creación y el manejo de las persistencias o beans.

Anexo 5: Instalación y configuración de la Base de Datos

Para la administración y gestión de la base de datos de este proyecto se utiliza la versión 5.6 de MySQL.

Pasos para la instalación de MySQL 5.6

- Descargamos de la versión descrita y su conector de la página oficial:
<http://dev.mysql.com/downloads/>
Versión: MySQL-5.6.26-2.sles12.x86_64.rpm-bundle.tar
Conector: mysql-connector-java-5.1.38.jar

La instalación se realiza en un equipo que será utilizado como servidor local el cual posee un Sistema Operativo Linux openSUSE Leap 42.1

Ejecutar el siguiente comando para la instalación de MySQL

```
sudo apt-get install mysql-server mysql-common mysql-client
```

Para iniciar los servicios de la Base de Datos de MySQL correr el siguiente comando:

```
sudo systemctl start
```

Comando para detener los servicios de la Base de Datos.

```
sudo systemctl stop
```


BIBLIOGRAFÍA

REFERENCIAS ELECTRÓNICAS:

Bibliografía

Red Hat, I. (18 de 03 de 2016). *JBOSS*. Obtenido de JBOSS: http://es.redhat.com/pdf/jboss/JBoss_Ent_app_platform_ES_web.pdf

MediaWiki, P. b. (19 de 03 de 2016). *Mercurial*. Obtenido de Mercurial: <https://es.wikipedia.org/wiki/Mercurial>

Raúl Monferrer Agust. (2001). *Especificación de Requisitos Software según el estándar de IEEE 830*. Mexico.

Group, C. ©.-2. (15 de 03 de 2016). *Documentación*. Obtenido de Documentación: <http://www.php.net/docs.php>

Oracle ©, O. C. (15 de 03 de 2016). *Manual de Referencia*. Obtenido de Manual de Referencia: <http://www.mysql.com/>

JavaServer™, F. (s.f.). *JavaServer Faces*. Recuperado el 02 de 05 de 2016, de <http://www.java-serverfaces.org/>

Schalk, C. (s.f.). *Introduction to Javasever Faces*. Recuperado el 06 de 05 de 2016, de Oracle Corporation: <http://www.oracle.com/technetwork/topics/index-090910.html>

Wikipedia. (s.f.). Recuperado el 07 de 06 de 2016, de Modelo–vista–controlador: <https://es.wikipedia.org/wiki/Modelo–vista–controlador>

Wikipedia. (s.f.). Recuperado el 07 de 07 de 2016, de JavaServer Faces: https://es.wikipedia.org/wiki/JavaServer_Faces

Informatics, P. (s.f.). *PrimeFaces*. Recuperado el 04 de 07 de 2016, de PrimeFaces: <http://www.primefaces.org>

Inc, P. (s.f.). Recuperado el 12 de 07 de 2016, de Que es PrimeFaces?: <https://prezi.com/nf2mkhghcndl/que-es-primefaces/>

Lerma, E. V. (s.f.). Recuperado el 15 de 07 de 2016, de Introducción a Primefaces: <https://www.adictosaltrabajo.com/tutoriales/introduccion-primefaces/>

Undercode. (s.f.). Recuperado el 20 de 07 de 2016, de Modelo Vista Controlador: <https://underc0de.org/foro/java/modelo-vista-controlador/>

Wikipedia. (s.f.). *WildFly*. Recuperado el 21 de 07 de 2016, de WildFly : <https://es.wikipedia.org/wiki/WildFly>

Wikipedia. (s.f.). *Enterprise JavaBeans*. Recuperado el 22 de 07 de 2016, de https://es.wikipedia.org/wiki/Enterprise_JavaBeans

affiliate., O. a. (s.f.). Recuperado el 15 de 06 de 2016, de The Java EE 5 Tutorial: <http://docs.oracle.com/javaee/5/tutorial/doc/bnblt.html>

Corporation, X. (s.f.). Recuperado el 16 de 06 de 2016, de What is AspectJ™: <http://web.archive.org/web/20000302211440/http://aspectj.org/servlets/AJSite>

Wikipedia. (s.f.). Recuperado el 10 de 06 de 2016, de Hibernate: <https://es.wikipedia.org/wiki/Hibernate>

community, M. (s.f.). *Mercurial*. Recuperado el 10 de 06 de 2016, de Mercurial: <https://www.mercurial-scm.org>

Mercurial, L. (s.f.). Recuperado el 16 de 06 de 2016, de Control Distribuido de Revisiones con Mercurial: <http://web.archive.org/web/20120701084001/http://devnull.li/libromercurial/index.es.html>

O'Sullivan, B. (s.f.). *The Definitive Guide*. Recuperado el 18 de 06 de 2016, de Mercurial: The Definitive Guide: <http://hgbook.red-bean.com>

Community, M. (s.f.). Recuperado el 20 de 07 de 2016, de Learning Mercurial in Workflows: <http://web.archive.org/web/20120510123141/http://mercurial.selenic.com/guide/>

Hat, R. (s.f.). Recuperado el 01 de 07 de 2016, de Wildfly 10: <http://www.wildfly.org/>

Elias, V. (s.f.). Recuperado el 10 de 07 de 2016, de Getting Started Guide: <https://docs.jboss.org/author/display/WFLY10/Getting+Started+Guide>

Foundation, F. S. (s.f.). Recuperado el 19 de 07 de 2016, de GNU Lesser General Public License: <https://www.gnu.org/licenses/lgpl-3.0.en.html>

©, L. C. (s.f.). Recuperado el 20 de 07 de 2016, de Ingenieria de Sistemas de Informacion: <http://es.slideshare.net/jpbthames/diseo-arquitectnico-9443843>

Corporation, L. (s.f.). Recuperado el 22 de 07 de 2016, de Diagramas de Despliegues: <http://es.slideshare.net/arcangelsombra/diagramas-de-despliegue-uml-1475353>

Doctora Jenny Ríos Coello, Secretaria de la Facultad de Ciencias de la Administración
de la Universidad del Azuay,

C E R T I F I C A:

Que, el Consejo de Facultad en sesión realizada el 29 de abril de 2016, conoció la solicitud de los señores DIEGO ALFREDO ENDERICA ORTEGA Y DIEGO FERNANDO PAUZHE HUGO con códigos 35571 y 37077 respectivamente, que presentan su trabajo de titulación denominado: **"DESARROLLO DE UNA APLICACIÓN DE FACTURACION ELECTRONICA UTILIZANDO HERRAMIENTAS DE GESTION E INTEGRACION EN ENTORNOS JAVA"** previa a la obtención del título de Ingenieros de Sistemas. El Consejo de Facultad acoge el informe de la Junta Académica y resuelve aprobar la denuncia. Designa como *Director al ingeniero Andrés Patiño León* y como *miembros del Tribunal Examinador a los ingenieros Juan Carlos Salgado y Rubén Ortega López*. De acuerdo al cronograma de la Unidad de Titulación Especial, los petitionarios para desarrollar y presentar su trabajo de titulación tienen un plazo de un período académico esto es **hasta el 09 de septiembre de 2016**, debiendo realizar dos sustentaciones parciales del trabajo de titulación: la semana entre el 30 de mayo al 03 de junio y la semana entre el 25 al 29 de julio de 2016 en su orden.

Cuenca, mayo 03 de 2016



Dra. Jenny Ríos Coello
Secretaria de la Facultad de
Ciencias de la Administración



UNIVERSIDAD DEL
AZUAY
FACULTAD DE
ADMINISTRACION
SECRETARIA



1.

CONVOCATORIA

Por disposición de la Junta Académica de la escuela de Ingeniería de Sistemas y Telemática, se convoca a los Miembros del Tribunal Examinador, a la sustentación del Protocolo del Trabajo de Titulación: **"Desarrollo de una aplicación de facturación electrónica utilizando herramientas de gestión e integración en entornos JAVA"**, presentado por los estudiantes **DIEGO ENDERICA ORTEGA** y **DIEGO FERNANDO PAUZHE** con códigos 35571 y 37077 respectivamente, previa a la obtención del grado de Ingenieros de Sistemas, para el Miércoles, 13 de abril de 2016 a las 17h00.

Cuenca, 08 de abril de 2016

Dra. Jenny Ríos Coello
Secretaria de la Facultad

Ing. Andrés Patiño León

Ing. Juan Carlos Salgado Arteaga

Ing. Rubén Ortega López

Comunicado 11/04/2016
D. 22pm

Cuenca, 20 de septiembre de 2016.- Decano de la Facultad de Ciencias de la Administración, con autorización amplia y suficiente concedida por el Consejo de Facultad en sesión del 25 de febrero de 2016, conoce la petición de los estudiantes **DIEGO ALFREDO ENDERICA ORTEGA** con código 35571 y **DIEGO FERNANDO PAUZHE HUGO**, con código 37077, quienes solicitan prórroga para la presentación del trabajo de titulación: *Desarrollo de una aplicación de facturación electrónica utilizando herramientas de gestión e integración en entornos JAVA*, presentado como requisito previo a la obtención del título de Ingeniero en de Sistemas, cuya fecha de la presentación venció el 09 de septiembre de 2016; *considerando el libera b) del Art. 6 del Instructivo para la Conformación y Funcionamiento de las Unidades de Titulación Especial en las Carreras de Grado de la Universidad del Azuay, aprobado por el Consejo Universitario el 5 de mayo de 2015, aprobar la solicitud y concede una prórroga seis de meses, esto es hasta el 09 de marzo de 2017.*

Cuenca, septiembre 20 de 2016.



Ing. Xavier Ortega Vásquez
Decano de la Facultad de
Ciencias de la Administración





1.



ACTA

SUSTENTACIÓN DE PROTOCOLO/DENUNCIA DEL TRABAJO DE TITULACIÓN

- 1.1 Nombre del estudiante: Diego Enderica Ortega y Diego Fernando Pauzhe
Códigos: 35571 y 37077 respectivamente
Director sugerido: Ing. Andrés Patriño León
- 1.2 Codirector (opcional): _____
- 1.3 Tribunal: Ing. Juan Carlos Salgado Arteaga e Ing. Rubén Ortega López
- 1.4 Título propuesto: *"Desarrollo de una aplicación de facturación electrónica utilizando herramientas de gestión e integración en entornos JAVA"*
- 1.5 Resolución:

1.5.1 Aceptado sin modificaciones _____

1.5.2 Aceptado con las siguientes modificaciones:

Retirar objetivo específico con numeral 1

Retirar el capítulo 3 y agregarlo como anexos.

Agregar un objetivo específico con las alternativas de integración.

1.5.3 Responsable de dar seguimiento a las modificaciones: Ing. Andrés Patriño.

1.5.4 No aceptado

• Justificación:

Tribunal

Ing. Andrés Patriño León

Ing. Juan Carlos Salgado

Ing. Rubén Ortega L.

Sr. Diego Enderica Ortega

S. Diego Fernando Pauzhe

Dra. Jenny Ríos Coello
Secretaria de Facultad

Fecha de sustentación: Miércoles, 13 de abril de 2016 a las 17H00.



RÚBRICA PARA LA EVALUACIÓN DEL PROTOCOLO DE TRABAJO DE TITULACIÓN

1.1 Nombre del estudiante: Diego Enderica Ortega y Diego Fernando Pazhe

Código 35571 y 37077 respectivamente

1.3 Director sugerido: Ing. Andrés Patiño León

1.1 Codirector (opcional):

1.2 Título propuesto: *"Desarrollo de una aplicación de facturación electrónica utilizando herramientas de gestión en integración en entornos JAVA"*

1.3 Revisores (tribunal): Ing. Juan Carlos Salgado e Ing. Rubén Ortega L.

1.4 Recomendaciones generales de la revisión:

	Cumple totalmente	Cumple parcialmente	No cumple	Observaciones (*)
Línea de investigación				
1. ¿El contenido se enmarca en la línea de investigación seleccionada?	✓			
Título Propuesto				
2. ¿Es informativo?	✓			
3. ¿Es conciso?	✓			
Estado del arte				
4. ¿Identifica claramente el contexto histórico, científico, global y regional del tema del trabajo?	✓			
5. ¿Describe la teoría en la que se enmarca el trabajo	✓			
6. ¿Describe los trabajos relacionados más relevantes?			✓	
7. ¿Utiliza citas bibliográficas?	✓			
Problemática y/o pregunta de investigación				
8. ¿Presenta una descripción precisa y clara?	✓			
9. ¿Tiene relevancia profesional y social?	✓			
Hipótesis (opcional)				
10. ¿Se expresa de forma clara?				
11. ¿Es factible de verificación?				
Objetivo general				
12. ¿Concuerda con el problema formulado?	✓			
13. ¿Se encuentra redactado en tiempo verbal infinitivo?	✓			
Objetivos específicos				

14. ¿Concuerdan con el objetivo general?	✓			
15. ¿Son comprobables cualitativa o cuantitativamente?	✓			
Metodología				
16. ¿Se encuentran disponibles los datos y materiales mencionados?	✓			
17. ¿Las actividades se presentan siguiendo una secuencia lógica?	✓			
18. ¿Las actividades permitirán la consecución de los objetivos específicos planteados?	✓			
19. ¿Los datos, materiales y actividades mencionadas son adecuados para resolver el problema formulado?	✓			
Resultados esperados				
20. ¿Son relevantes para resolver o contribuir con el problema formulado?	✓			
21. ¿Concuerdan con los objetivos específicos?	✓			
22. ¿Se detalla la forma de presentación de los resultados?	✓			
23. ¿Los resultados esperados son consecuencia, en todos los casos, de las actividades mencionadas?	✓			
Supuestos y riesgos				
24. ¿Se mencionan los supuestos y riesgos más relevantes?	✓			
25. ¿Es conveniente llevar a cabo el trabajo dado los supuestos y riesgos mencionados?	✓			
Presupuesto				
26. ¿El presupuesto es razonable?	✓			
27. ¿Se consideran los rubros más relevantes?	✓			
Cronograma				
28. ¿Los plazos para las actividades son realistas?	✓			
Referencias				
29. ¿Se siguen las recomendaciones de normas internacionales para citar?	✓			
Expresión escrita				
30. ¿La redacción es clara y fácilmente comprensible?	✓			
31. ¿El texto se encuentra libre de faltas ortográficas?	✓			

(*) Breve justificación, explicación o recomendación.

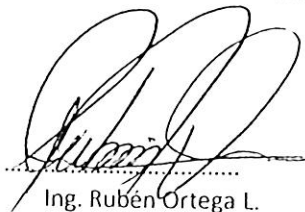
- Opcional cuando cumple totalmente,
- Obligatorio cuando cumple parcialmente y NO cumple.



Ing. Andrés Patiño León



Ing. Juan Carlos Salgado



Ing. Rubén Ortega L.

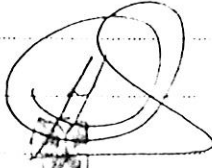
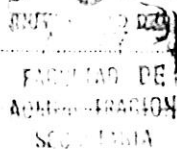


DOCTORA JENNY RIOS COELLO, SECRETARIA DE LA FACULTAD DE
CIENCIAS DE LA ADMINISTRACION DE LA UNIVERSIDAD DEL AZUAY

CERTIFICA:

Que, el señor **ENDERICA ORTEGA DIEGO ALFREDO**, con código 35571, luego de
cumplir con todas las asignaturas del pensum de la Escuela de **INGENIERIA DE
SISTEMAS**, finalizó sus estudios en la Facultad el 11 de Marzo de 2016.

Cuenca, 06 de abril de 2016

Derecho No. 001-010-000098435

mjmr.-

Cuenca, 08 de abril de 2016

Señor Ingeniero

Xavier Ortega Vásquez

DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACION

Presente.-

De nuestras consideraciones:

Nosotros, Diego Alfredo Enderica Ortega con código 35571 y Diego Fernando Pauzhe

Hugo con código 37077, egresados de la escuela de Ingeniería de Sistemas de la

Facultad de Ciencias de la Administración, solicitamos de la manera mas comedida nos

apruebe el tema de tesis titulado "Desarrollo de una aplicación de facturación

electrónica utilizando herramientas de gestión e integración en entornos Java" previo a

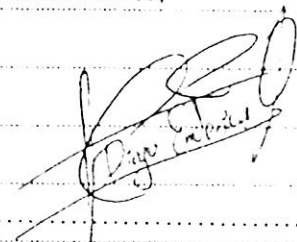
la obtención del título de Ingenieros de Sistemas, el cual ya ha sido revisado por el Ing.

Andrés Patiño.

Ante la favorable acogida que de a la presente anticipamos nuestros mas sinceros

agradecimientos.

Atentamente,



Diego Enderica Ortega.

C.I. 0105247076

Código: 35571



Diego Pauzhe Hugo

C.I. 0302168349

Código. 37077

GUÍA PARA LA ELABORACION Y PRESENTACION DE LA DENUNCIA/PROTOCOLO DE TRABAJO DE TITULACIÓN

1. DATOS GENERALES

1.1 Nombres:

1.1.1 Enderica Ortega Diego Alfredo

1.1.2 Código: 35571

1.1.3 Contacto: 4085981,0984485661, diegoende@hotmail.com

1.1.4 Pazuhe Hugo Diego Fernando

1.1.5 Código: 37077

1.1.6 Contacto: 0988632243, nandop845@hotmail.com

1.2 Director Sugerido: Ing. Andrés Patiño

1.2.1 Contacto: andpatino@uazuay.edu.ec

1.3 Co. director: No.

1.4 Asesor Metodológico: No.

1.5 Tribunal Designado: Pendiente.

1.6 Aprobación: Pendiente.

1.7 Línea de Investigación de la carrera.

1.7.1 UNESCO: 1203 Informática de computadores

1.7.2 Tipo de trabajo: 1203.18 Sistemas de Información.- Desarrollo modular de un sistema de planificación de recursos empresariales ERP (Enterprise Resource Planning).

1.8 Área de Estudio

Programación Orientada a Objetos

1.9 Título Propuesto

Desarrollo de una aplicación de Facturación Electrónica utilizando herramientas de gestión e integración en entornos Java.

1.10 Subtítulo.

1.11 Estado del Proyecto

El presente proyecto se propone demostrar las ventajas de utilizar las herramientas Maven, Hibernate, Java Server Faces (JSF), Prime Faces y MySQL dentro del entorno de desarrollo, permitiéndonos integrar, gestionar y administrar de mejor manera nuestros proyectos, para lo cual se implementará una aplicación para emisión de facturas electrónicas.

2. CONTENIDO

2.1 Motivación de la investigación

Hoy en día en nuestro medio, de acuerdo a nuevas disposiciones establecidas por el Servicio de Rentas Internas (SRI), las empresas deben de manera obligatoria completar el esquema de emisión de comprobantes electrónicos.

Este medio supone grandes ventajas, como por ejemplo: menor costo en el cumplimiento de obligaciones tributarias, mayor seguridad en el resguardo de documentos, disminución del trabajo de almacenamiento físico, cuidado del medio ambiente y consultas en línea de documentos electrónicos, etc.

En consecuencia, además de la finalidad de alcanzar nuestro título de grado, nos hemos visto motivados a realizar este proyecto, para poder demostrar las ventajas de las herramientas utilizadas en el entorno de desarrollo, y a la vez contribuir de alguna manera a empresas que requieren de este tipo de servicio.

2.2 Problemática

Entre algunos de los problemas que se presentan dentro del desarrollo de aplicaciones con Java tenemos:

- La integración de módulos y proyectos es compleja consumiendo gran cantidad de tiempo y esfuerzo

- Al desarrollar en modelos orientados a objetos, generalmente se reutilizan gran cantidad de clases externas, las cuales a su vez dependen de otras adicionales. Esta gestión introduce mayor complejidad en la programación
- La gestión del proyecto se vuelve complicada debido a que cada miembro del equipo de desarrollo debe estar instalando el código fuente en cada uno de sus equipos en los que se programa, causando de esta manera que algunos miembros lleven el código desactualizado, haciendo tediosa la integración de los módulos del proyecto, produciendo de esta manera una inadecuada estimación en los tiempos de duración de las tareas y un bajo rendimiento en la funcionalidad del producto final.
- Se presentan grandes inconvenientes en el desarrollo de software si no se posee un sistema de control de versiones, no se podrían controlar los cambios que se van realizando por parte de los miembros del equipo de desarrollo.
- Al interactuar con la base de datos generalmente creamos clases para realizar un insert o un update lo que conlleva a reescribir código en lugar de reutilizarlo. Al tener clases que funcionan de manera exclusiva en ciertas bases de datos hace que se tengan que reprogramar para que cumplan su función con otras base de datos diferentes.

Como solución a estas problemáticas mostramos las funcionalidades que presentan Maven e Hibernate como herramientas de integración, gestión y administración del proyecto.

Maven nos simplifica el uso de librerías permitiéndonos gestionar nuestros recursos de una mejor manera, así como para el resto de desarrolladores que están inmersos en el proyecto pueden acceder al repositorio y estar actualizados en la última versión del mismo. Gracias a esta herramienta de software, el proyecto puede ser compatible con cualquier otro proyecto, ya que tiene un modelo de configuración y de construcción más simple, basado en un formato XML.

Con Hibernate se facilita la interacción con la base de datos de nuestro proyecto; nos facilita la gestión con la misma y nuestro proyecto puede adaptarse a cualquier base de datos gracias a esta herramienta.

Por otra parte, la aplicación pretende solventar la problemática de la implementación obligatoria de la emisión de facturas electrónicas cuya fecha máxima para contribuyentes especiales es el 31 de diciembre de 2017.

2.3 Pregunta de Investigación

¿Las características que ofrecen Maven, Hibérnate, Java Server Faces JSF, Prime Faces y MySQL como entorno de desarrollo facilitan la integración, gestión y administración de proyectos JAVA ?

2.4 Resumen

El propósito de este Proyecto de Titulación consiste en implementar una aplicación para emitir facturas electrónicas, utilizando Maven, Hibérnate, Java Server Faces (JSF), Prime Faces y MySQL como herramientas de desarrollo e integración. Se hace énfasis en Hibernate y Maven porque son consideradas como dos de las mejores herramientas para desarrollo software con JAVA y a la vez, son poco conocidas en nuestro medio.

Hibernate es un potente servicio de persistencia Objeto – Relacional de alto rendimiento que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación. Ofrece persistencia automatizada y transparente de objetos a tablas en una base de datos relacional, utilizando metadata que describe el mapeo entre objetos y la base de datos en si.

Entre las principales ventajas de Hibernate están:

- Productividad: Evita mucho del código confuso en la capa de persistencia, permitiendo centrarse en la lógica de negocio.

- Fácil mantenimiento: Por tener pocas líneas de código, permite que el código sea más claro. Al dividir la capa de persistencia, se puede identificar los errores muy fácilmente.

- Rendimiento: Hibernate tiene un buen desempeño, el mismo que está vinculado a la estructura de las consultas y la configuración del Framework en la aplicación o proyecto.

Cuando tenemos un proyecto Java, normalmente hay muchas tareas que poco tienen que ver con la aplicación que estamos creando, pero sí con el entorno de trabajo. Siempre debemos crear la estructura de directorios (src, resources, lib...), configurar los accesos a bases de datos y a otros repositorios, comprobar que existe o configurar un directorio para depositar los ficheros compilados (.class), los .jar, etc. Todos estos inconvenientes se pueden solucionar gracias a **Maven**, que es una herramienta capaz de generar todas las estructuras de directorios para nuestro proyecto, con Maven podremos descargar las librerías que nos hagan falta en nuestro proyecto.

2.5 Estado del Arte y Marco Teórico

Antecedentes

En la actualidad el uso de aplicaciones sigue siendo un sector en expansión, en el que convertimos nuestro entorno digital en nuestro día a día, lo cual representa una gran ventaja para los que estamos involucrados en el ámbito de desarrollo de aplicaciones, ya que en nuestro medio aumenta el mercado que necesita sistemas informáticos.

A consecuencia de esto y como una posible solución a la necesidad de la facturación electrónica, nosotros realizaremos una aplicación en un entorno de desarrollo utilizando las herramientas propuestas, las cuales nos permiten integrar y gestionar de una mejor manera el proceso de desarrollo, facilitándonos la integración del proyecto.

Bases teóricas

Para el desarrollo de esta aplicación es necesario tener conocimientos de las siguientes herramientas:

- Maven
- Hibernate
- Java Server Faces JSF
- PrimeFaces
- Base de datos MySQL
- Jboss
- Mercurial

Maven. Es una herramienta de software para la gestión y construcción de proyectos Java. Maven utiliza un Project Object Model (POM) para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externos 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. (Foundation, 2016).

Una característica clave de Maven es que está listo para usar en red. El motor incluido en su núcleo puede dinámicamente descargar plugins de un repositorio, el mismo que provee acceso a muchas versiones de diferentes proyectos Open Source en Java, de Apache y otras organizaciones y desarrolladores. Este repositorio y su sucesor reorganizado, el repositorio Maven 2, pugnan por ser el mecanismo de facto de distribución de aplicaciones en Java, pero su adopción ha sido muy lenta.

Maven provee soporte no solo para obtener archivos de su repositorio, sino también para subir artefactos al repositorio al final de la construcción de la aplicación, dejándola al acceso de todos los usuarios. Una caché local de artefactos actúa como la primera fuente para sincronizar la salida de los proyectos a un sistema local. (Fundación Wikimedia, 2016)

Hibernate.- Hibernate es una herramienta de mapeo objeto/relacional para entornos Java. El término de mapeo objeto/relacional (ORM) se refiere a la técnica de mapear una representación de datos desde un modelo de objeto a un modelo de datos relacionales con un esquema basado en SQL.

Hibernate no solamente se ocupa del mapeo desde las clases Java a las tablas de las bases de datos (y desde los tipos de datos de Java a los tipos de datos de SQL), sino que también facilita la consulta y recuperación de datos. Esto puede reducir de manera importante el tiempo de desarrollo que se tomaría con el manejo de datos de forma manual en SQL y JDBC.

La meta de Hibernate es aliviar el trabajo del desarrollador en 95% de la persistencia de datos comunes relacionados con tareas de programación. (Red Hat I., 2016).

Como todas las herramientas de su tipo, Hibernate busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen. Con esta información Hibernate le permite a la aplicación manipular los datos en la base de datos operando sobre objetos, con todas las características de la POO. Hibernate convertirá los datos entre los tipos utilizados por Java y los definidos por SQL. Hibernate genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todos los motores de bases de datos con un ligero incremento en el tiempo de ejecución.

Hibernate está diseñado para ser flexible en cuanto al esquema de tablas utilizado, para poder adaptarse a su uso sobre una base de datos ya existente. También tiene la funcionalidad de crear la base de datos a partir de la información disponible. Hibernate ofrece también un lenguaje de consulta de datos llamado HQL (Hibernate Query Language), al mismo tiempo que una API para construir las consultas programáticamente.

Java Server Faces JSF.- Java Server Faces (JSF) es un framework de aplicaciones web basado en Java destinada a simplificar la integración del desarrollo basado en la web de interfaces de usuario. (Simon, 2009)

JSF incluye:

- Un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar eventos, validar entradas, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad.
- Un conjunto por defecto de componentes para la interfaz de usuario.
- Dos bibliotecas de etiquetas personalizadas para Java Server Pages que permiten expresar una interfaz Java Server Faces dentro de una página JSP.
- Un modelo de eventos en el lado del servidor.
- Administración de estados.
- Beans administrativos.
- Java Server Faces es una tecnología de visualización normalizada que se formalizó en una especificación a través del Java Community Process

Versiones de JSF:

- JSF 1.0 (11-03-2004). lanzamiento inicial de las especificaciones de JSF.

- JSF 1.1 (27-05-2004). lanzamiento que solucionaba errores. Sin cambios en las especificaciones ni en el render kit de HTML.
- JSF 1.2 (11-05-2006). lanzamiento con mejoras y corrección de errores.
- JSF 2.0 (12-08-2009). último lanzamiento.

Prime Faces.- Es una suite open source de componentes que extiende a JSF. Prime Faces es una librería de componentes visuales open source desarrollada y mantenida por Prime Technology, una compañía Turca de IT especializada en consultoría ágil, JSF, Java EE y Outsourcing. El proyecto es liderado por Çağatay Çivici, un miembro del “JSF Expert Group”

Las principales características de Primefaces son:

- Soporte nativo de Ajax, incluyendo Push/Comet.
- Kit para crear aplicaciones web para móviles.
- Es compatible con otras librerías de componentes, como JBoss RichFaces.
- Uso de javascript no intrusivo (no aparece en línea dentro de los elementos, sino dentro de un bloque <script>).

Base de Datos MySQL.- “Es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca.

MySQL es un software de código abierto, licenciado aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario. (Oracle ©, 2016)

El lenguaje de programación que utiliza MySQL es Structured Query Language (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales”. (Group, 2016)

Jboss.- Es un servidor de aplicaciones Java EE de código abierto implementado en Java puro. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo para el que esté disponible la máquina virtual de Java. (Red Hat I. , 2016)

JBoss Enterprise Application Platform es la plataforma para aplicaciones Java líder en la industria, integrada, simplificada y proporcionada por el líder en software de código abierto para empresas.

Integrando las tecnologías punteras del mercado en una única y sencilla solución, la Plataforma de Aplicaciones JBoss Enterprise facilita el desarrollo, la implantación y la gestión de las aplicaciones Java. Incluye las tecnologías de código abierto más avanzadas para la creación, despliegue y alojamiento de aplicaciones Java de empresas y de servicios en una arquitectura orientada a servicios. Con la Plataforma de Aplicaciones JBoss Enterprise, las aplicaciones Java para empresa son sencillas, abiertas y asequibles.

Mercurial.- Es un sistema de control de versiones multiplataforma, para desarrolladores de software. Está implementado principalmente haciendo uso del lenguaje de programación Python, pero incluye una implementación binaria de diff escrita en C. Mercurial fue escrito originalmente para funcionar sobre GNU/Linux. Ha sido adaptado para Windows, Mac OS X y la mayoría de otros sistemas tipo Unix. Mercurial es, sobre todo, un programa para la línea de comandos. Todas las operaciones de Mercurial se invocan como opciones dadas a su programa motor, hg (cuyo nombre hace referencia al símbolo químico del mercurio). (MediaWiki, 2016)

Las principales metas de desarrollo de Mercurial incluyen un gran rendimiento y escalabilidad; desarrollo completamente distribuido, sin necesidad de un servidor; gestión robusta de archivos tanto de texto como binarios; y capacidades avanzadas de ramificación e integración, todo ello manteniendo sencillez conceptual. Incluye una interfaz web integrada.

2.6 Hipótesis

No

2.7 Objetivo General

Desarrollar una aplicación para generar Facturas Electrónicas utilizando Maven, Hibernate, Java Server Faces JSF, Prime Faces, MySQL como herramientas de integración, gestión y administración del proyecto.

2.8 Objetivos Específicos

1. Integrar las herramientas planteadas formando un entorno de desarrollo de software para nuestro proyecto.
2. Implementar una aplicación para emitir la factura electrónica, la cual se debe firmar, enviar y consultar el servicio Web del SRI.
3. Generar el RIDE del documento electrónico.
4. Definir formatos de intercambio de datos que nos permitirá la integración con otros aplicativos

2.9 Metodología

Para el desarrollo de este proyecto que hemos planteado y al mismo tiempo cumplir con los objetivos propuestos, se ha seguido una metodología que consta de los siguientes puntos.

TIPO DE ESTUDIO

El desarrollo de este proyecto de titulación se lo realizará de forma aplicada, de manera que llevaremos a cabo la implementación de un sistema que genere facturas electrónicas, demostrando el uso de las herramientas descritas en el entorno de desarrollo y cómo afectan a la gestión e integración del proyecto.

MÉTODO

Se utilizará para el desarrollo de este proyecto de titulación el método Analítico – Sintético, ya que se estudiarán y analizarán varias herramientas necesarias para el desarrollo de esta aplicación.

MEDIOS

Las fuentes para la recopilación de información serán las siguientes:

- Bibliografía.
- Internet.
- Libros electrónicos.
- Artículos relacionados publicados en Internet.
- Experiencia personal.

2.10 Alcances y resultados Esperados

- Demostrar la integración y funcionamiento correcto de las herramientas utilizadas en este proyecto.
- Justificar la implementación del entorno de desarrollo propuesto
- Desarrollo del aplicación para la facturación electrónica
- Demostrar la funcionalidad de la aplicación a través de la emisión e impresión de facturas electrónicas.

2.11 Supuestos y Riesgos

En la realización de este proyecto de titulación un supuesto riesgo sería la falta de una firma electrónica autorizada por el Banco Central del Ecuador, para llevar a cabo las pruebas necesarias para firmar el documento electrónico. Para solucionar este supuesto riesgo hemos conseguido un RUC con el que vamos a comprar una firma electrónica de pruebas en el Banco Central; de esta manera podremos realizar las pruebas necesarias de nuestra aplicación.

Otro posible riesgo que se nos puede presentar y causar inconvenientes o retraso en el desarrollo del proyecto es la falta de disponibilidad del servicio Web del SRI. La solución para este riesgo es generar un proceso en el cual el envío se realice mediante hilos (Clase *Thread*) de esta manera se enviara internamente hasta que tenga disponibilidad el SRI en modo OFF-LINE

2.12 Presupuesto

Rubro - denominación	Costo USD	Justificación ¿Para qué?
Equipo portátil MAC	\$ 1000	Dispositivo tecnológico para investigación de las herramientas, y desarrollo de la aplicación
Equipo portátil TOSHIBA	\$ 1200	Dispositivo tecnológico para investigación de las herramientas, y desarrollo de la aplicación
Router inalámbrico	\$ 30	Proporcionar conexión a los computadores
Conexión a internet	\$ 120	Para realizar la investigación y consulta de herramientas de desarrollo
Gastos de impresión de tesis	\$ 200	Impresión de documentación de tesis
Firma Digital	\$ 30	Firma digital para realizar pruebas del documento electrónico.
Imprevistos	\$ 100	Gastos varios
Transporte	\$ 100	Movilización de los miembros del equipo.
TOTAL	2780	TOTAL

2.13 Financiamiento

La financiación de este proyecto está a cargo de los autores, Enderica Ortega Diego Alfredo y Pazhe Hugo Diego Fernando.

2.14 Esquema Tentativo

Introducción

1. Capítulo 1: Tecnologías aplicadas

1.1 Herramientas de desarrollo

1.1.1 Maven

1.1.2 Hibernate

1.1.3 Java Server Faces JSF

1.1.4 Prime Faces

1.1.5 JBoss

1.1.6 Mercurial

1.2 Base de Datos

1.2.1 MySQL

2. Capítulo 2: Análisis y Diseño del Sistema

2.1. Especificación de Requisitos de Software (ERS) (Raúl Monferrer Agust, 2001)

2.1.1. Levantamiento de requisitos

2.2. Diseño de Datos.

2.2.1. Diagrama de Entidad - Relación

2.2.2. Diccionario de Datos

2.3. Diseño de Procesos.

2.3.1. Diagrama de Actividades.

2.4. Diseño Arquitectónico.

2.4.1. Diagrama de despliegue

2.4.2. Diseño de interfaces.

3. Capítulo 3: Desarrollo de la aplicación.

3.1 Creación de mantenimientos indispensables.

3.2 Creación de las clases necesarias para construir el XML de la factura electrónica.

- 3.3 Creación de las clases que firma el XML.
- 3.4 Creación de las clases que envía y consulta el estado del comprobante electrónico.
- 3.5 Generar el RIDE para la impresión del comprobante.
- 3.6 Pruebas del aplicativo
 - 3.6.1 Pruebas de funcionamiento
 - 3.6.2 Pruebas de integración
 - 3.6.3 Pruebas de rendimiento
- 4. Conclusiones y Recomendaciones
- 5. Anexos
 - 5.1 Integración de las herramientas de desarrollo
 - 5.2 Instalación del servidor de aplicaciones JBoss
 - 5.3 Instalación del administrador de proyectos Maven
 - 5.4 Instalación de Hibernate.
 - 5.5 Instalación y configuración de la Base de Datos

2.15 Cronograma

ACTIVIDADES	MES 1	MES 2	MES 3	MES 4
DESCRIBIR LAS HERRAMIENTAS DE DESARROLLO				
JUSTIFICAR LA IMPLMENTACION DEL ENTORNO DE DESARROLLO				
ESPECIFICACIÓN DE REQUISITOS DEL SOFTWARE				
DISEÑO DE DATOS				
DISEÑO DE PROCESO				
DISEÑO ARQUITECTÓNICO				

[illegible]

Objetivo Específico	Actividad	Resultado Esperado	Tiempo(Semanas)
Describir las herramientas de desarrollo	Describir conceptos fundamentales	Obtener un documento con los conceptos e incrementar nuestro conocimientos de las herramienta	1
	Justificar la implementación del entorno de desarrollo	Demostrar las funciones del entorno de desarrollo implementado	
Análisis y diseño del sistema	Levantamiento de requisitos	Obtener los requisitos necesarios de acuerdo a las necesidades de la aplicación.	1



UNIVERSIDAD DEL
AZUAY

	Realizar el diseño de datos	Modelos de datos	1
	Diseño de procesos.	Obtener el diagrama de Actividad para cada proceso.	2
	Diseño Arquitectónico	Obtener modelo de despliegue y diseño de interfaz	1
Integración de las herramientas de desarrollo	Integración.	Obtención del entorno de desarrollo	2
	Creación de mantenimientos indispensables	Implementar los mantenimientos	2

	Creación de las clases necesarias para construir el XML	Obtención del archivo XML	2
	Creación de las clases que firman el XML	Obtención del archivo XML firmado electrónicamente.	2
	Creación de las clases que envía y consulta el estado del comprobante	Obtención del estado del comprobante	1
	Generar del RIDE para la impresión del comprobante	Obtención del RIDE	
Pruebas de la aplicación.	Realizar pruebas de funcionamiento de la aplicación	Verificar de la funcionalidad del sistema	1
TOTAL SEMANAS			16

2.17 Referencias

Bibliografía

Raúl Monferrer Agust. (2001). *Especificación de Requisitos Software según el estándar de IEEE 830*. Mexico.

Foundation, T. A. (25 de 01 de 2016). *Apache Maven Project*. Obtenido de Apache Maven Project: <https://maven.apache.org>

Fundación Wikimedia, I. (20 de 02 de 2016). *Maven*. Obtenido de Maven: <https://es.wikipedia.org/wiki/Maven>

Red Hat, I. (22 de 02 de 2016). *HIBERNATE*. Obtenido de HIBERNATE: https://docs.jboss.org/hibernate/orm/3.5/reference/es-ES/html_single/

Simon, C. (08 de 10 de 2009). *JavaServer Faces*. Obtenido de JavaServer Faces: http://www.solucionjava.com/pdf/Curso_JSF2_Hibernate3.pdf

Red Hat, I. (18 de 03 de 2016). *JBOSS*. Obtenido de JBOSS: http://es.redhat.com/pdf/jboss/JBoss_Ent_app_platform_ES_web.pdf

Group, C. ©.-2. (15 de 03 de 2016). *Documentación*. Obtenido de Documentación: <http://www.php.net/docs.php>

MediaWiki, P. b. (19 de 03 de 2016). *Mercurial*. Obtenido de Mercurial: <https://es.wikipedia.org/wiki/Mercurial>

Oracle ©, O. C. (15 de 03 de 2016). *Manual de Referencia*. Obtenido de Manual de Referencia: <http://www.mysql.com/>

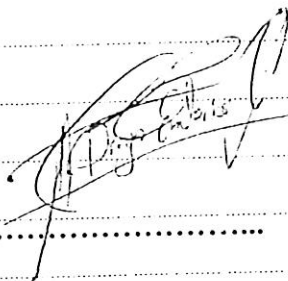
Sonatype, I. (01 de 01 de 2008). *Maven: The Complete Reference*. *Maven: The Complete Reference*. (I. Sonatype, Ed.)

Shah, S. (2014). *Maven para Eclipse*. Packt Publishing Ltd.

Jeff Linwood, D. M. (2010). *Beginning Hibernate*. (Apress, Ed.)

2.17 Anexos: para casos en los que se requiera respaldar el proyecto.

2.18 Firma de responsabilidad (estudiantes)



Diego Enderica O.



Diego Pazhe H.

Código: 35571

Código: 37077

2.19 Firma de responsabilidad (director sugerido)



Ing. Andrés Patiño

2.20 Fecha de entrega: 6 de abril de 2016