



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
E INFORMÁTICOS**

TEMA:

**“MÓDULO DE GESTIÓN DE INVENTARIO Y FACTURACIÓN BAJO
ARQUITECTURA SOA PARA EL SISTEMA ADMINISTRATIVO
CONTABLE DE LA EMPRESA BESIXPLUS CIA. LTDA.”**

Trabajo de Graduación Modalidad TEMI. Trabajo Estructurado de Manera Independiente, presentado previo la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

AUTOR: ANA VIRGINIA FLORES PIMBO

TUTOR: ING. CLAY ALDÁS

AMBATO – ECUADOR

2012

APROBACIÓN DEL TUTOR

En mi calidad de tutor del Trabajo Estructurado de Manera Independiente (TEMI) sobre el tema: **“MÓDULO DE GESTIÓN DE INVENTARIO Y FACTURACIÓN BAJO ARQUITECTURA SOA PARA EL SISTEMA ADMINISTRATIVO CONTABLE DE LA EMPRESA BESIXPLUS CIA. LTDA.”**, de la señorita **Flores Pimbo Ana Virginia**, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos , de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad el Art. 16 del Capítulo II, del Reglamento de Graduación para Obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

Ambato enero, 2012

EL TUTOR

Ing. Clay Aldás

AUTORÍA

El presente trabajo de investigación titulado: “**MÓDULO DE GESTIÓN DE INVENTARIO Y FACTURACIÓN BAJO ARQUITECTURA SOA PARA EL SISTEMA ADMINISTRATIVO CONTABLE DE LA EMPRESA BESIXPLUS CIA. LTDA.**” Es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato enero, 2012

Ana Virginia Flores Pimbo

CC: 1804192837

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Dr. Mauricio Carranza y el Ing. Francisco López, revisó y aprobó el Informe Final del trabajo de graduación titulado “**MÓDULO DE GESTIÓN DE INVENTARIO Y FACTURACIÓN BAJO ARQUITECTURA SOA PARA EL SISTEMA ADMINISTRATIVO CONTABLE DE LA EMPRESA BESIXPLUS CIA. LTDA.**” presentado por la señorita Ana Virginia Flores Pimbo, de acuerdo al Art. 18 del Reglamento de Graduación para obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

Ing. Oswaldo Paredes
PRESIDENTE DEL TRIBUNAL

Dr. Mauricio Carranza
DOCENTE CALIFICADOR

Ing. Francisco López
DOCENTE CALIFICADOR

DEDICATORIA

Este trabajo de investigación va dedicado a todas las personas que confiaron en mí en especial a mi madre y padre quienes han iluminado el sendero por donde camino, a mis hermanos, amigos y profesores sin los cuales la carga hubiera sido demasiada y a DIOS cuyas bendiciones fueron infaltables al momento de enfrentar los obstáculos.

Ana Virginia Flores Pimbo

AGRADECIMIENTO

En primer lugar, a Dios, por darme la vida y colmarla de bendiciones.

A mis padres, familiares y amigos, por todo su amor, comprensión y apoyo incesable.

A mis profesores, quienes han brindado sus conocimientos y hoy pueden ver un reflejo de lo que han formado.

A mi profesor guía, Clay Aldás, quien con sus consejos supo guiar y motivar en el desarrollo del trabajo de graduación.

A la empresa BESIXPLUS CIA. LTDA., la cual abrió sus puertas e hizo posible la elaboración de este trabajo.

Ana Virginia Flores Pimbo

ÍNDICE

Carátula	i
Aprobación del tutor	ii
Autoría.....	iii
Aprobación de la comisión calificadora.....	iv
Dedicatoria	v
Agradecimiento	vi
Índice	vii
Índice de figuras	xii
Índice de tablas.....	xv
Resumen Ejecutivo.....	xvii
Introducción	xviii

CAPÍTULO I

EL PROBLEMA DE INVESTIGACIÓN

1.1 Tema.....	1
1.2 Planteamiento del problema.....	1
1.2.1 Contextualización.....	1
1.2.2 Análisis Crítico.....	2
1.2.3 Prognosis	2
1.3 Formulación del problema	3
1.3.1 Preguntas directrices	3
1.3.2 Delimitación del objeto de investigación	4
1.4 Justificación.....	4
1.5 Objetivos	5
1.5.1 Objetivo General	5
1.5.2 Objetivos Específicos.....	5

CAPÍTULO II

MARCO TEÓRICO

2.1 Antecedentes Investigativos	6
2.2 Fundamentación legal	7
2.2.1 Empresa BESIXPLUS CIA. LTDA.	7
2.3 Categorías Fundamentales	9
2.3.1 Servidor de base de datos.....	11
2.3.1.1 PostgreSQL	12
2.3.2 Servidor de aplicaciones	14
2.3.2.1 Jboss	15
2.3.3 Jasper Server	15
2.3.4 Plataforma JEE (Java Enterprise Edition).....	16
2.3.4.1 Herramientas de desarrollo JEE	17
2.3.4.2 Lenguaje de programación Java.....	17
2.3.5 SOA.....	19
2.3.6 Servicios web	21
2.3.7 Tecnologías, lenguajes y librerías web del lado del cliente	22
2.3.7.1 AJAX (Asynchronous Javascript y XML).....	22
2.3.7.2 HTML	23
2.3.7.3 DHTML	23
2.3.7.4 JavaScript	23
2.3.7.5 XML (Extensible Markup Lenguaje).....	24
2.3.7.6 CSS (Cascading Style Sheets).....	24
2.3.7.7 DOM (Document Object Model)	24
2.3.7.8 Ext Js	25
2.3.8 Navegadores web	25
2.3.9 Inventarios	26
2.3.10 Facturación	27
2.3.11 Administración	28
2.3.12 Contabilidad	28
2.4 Hipótesis	28
2.5 Señalamiento de las variables de la hipótesis	28
2.5.1 Variable Independiente	28

2.5.2 Variable Dependiente.....	28
---------------------------------	----

CAPÍTULO III METODOLOGÍA

3.1 Enfoque	29
3.2 Modalidad de la investigación	29
3.2.1 Investigación de campo.....	29
3.2.2 Investigación bibliográfica - documental.....	29
3.2.3 Proyecto factible.....	30
3.3 Nivel o tipo de investigación.....	30
3.4 Población y muestra	30
3.4.1 Población	30
3.4.2 Muestra.....	30
3.5 Operacionalización de las variables	31
3.5.1 Variable independiente.....	31
3.5.1 Variable dependiente.....	32
3.6 Recolección de Información	33
3.6.1 Plan de recolección de información	33
3.6.2 Plan de procesamiento y análisis de la información	33

CAPÍTULO IV ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

4.1 Análisis e interpretación de los resultados	34
4.1.1 Análisis de los resultados de las entrevistas.....	34

CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones	45
5.2 Recomendaciones.....	46

CAPÍTULO VI PROPUESTA

6.1 Datos Informativos	47
6.2 Antecedentes de la propuesta	47
6.3 Justificación.....	48
6.4 Objetivos	48
6.4.1 Objetivo General	48
6.4.2 Objetivos Específicos.....	49
6.5 Análisis de factibilidad	49
6.5.1 Factibilidad Operativa.....	49
6.5.2 Factibilidad Económica	50
6.5.3 Factibilidad Técnica	50
6.6 Fundamentación	51
6.7 Metodología	52
6.8 Modelo Operativo	53
6.8.1 Análisis del Sistema	53
6.8.1.1 Análisis y requerimientos del sistema.....	53
6.8.1.2 Diagramas UML.....	59
6.8.2 Diagrama de despliegue	83
6.8.3 Diseño del sistema.....	84
6.8.3.1 Arquitectura de software	84
6.8.3.2 Diseño de la base de datos	87
6.8.3.3 Diccionario de datos.....	89
6.8.4 Implementación.....	104
6.8.5 Implantación.....	125
6.8.6 Pruebas	126
6.8.6.1 Pruebas unitarias	126
6.8.6.2 Pruebas de caja blanca	128
6.8.6.3 Pruebas de caja negra	129
6.9 Conclusiones y recomendaciones	131

6.9.1 Conclusiones	131
6.9.2 Recomendaciones.....	132
6.10 Glosario de términos	133
6.11 Bibliografía	137
Anexos	

ÍNDICE DE FIGURAS

CAPÍTULO II MARCO TEÓRICO

Figura 2.1. Categorías fundamentales.....	9
Figura 2.2. Logo de PostgreSQL.....	12
Figura 2.3. Logo de JBoss.....	15
Figura 2.4. Interrelación entre los principios de SOA	21
Figura 2.5. Proceso de uso de un Servicio Web.....	22
Figura 2.6. Logo de ExtJS.....	25
Figura 2.7. Logo de Google Chrome, Mozilla Firefox e Internet Explorer.....	25
Figura 2.8. Kardex de inventario.....	27

CAPÍTULO IV ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

Figura 4.1. Sistema de inventario utilizado.....	35
Figura 4.2. Forma de control de mercadería	36
Figura 4.3. Procedimiento para elaborar facturas	37
Figura 4.4. Conocimientos de reglamentos del SRI.....	38
Figura 4.5. Ventajas de utilizar el sistema a través de internet.....	39
Figura 4.6. Servicio al cliente	40
Figura 4.7. Almacenamiento de la información.....	41
Figura 4.8. Manejo de paquetes contables	42
Figura 4.9. Manual de funciones.....	43
Figura 4.10. Crecimiento de ventas en la Empresa.....	44

CAPÍTULO VI PROPUESTA

Figura 6.1. Metodología XP.....	52
Figura 6.2. Diagrama de caso de uso: Interacción del Módulo de gestión de inventario y facturación y los módulos de clientes, compras y contabilidad.....	60
Figura 6.3. Diagrama de caso de uso: de Gestión de inventario y facturación.....	60
Figura 6.4. Diagrama de caso de uso: Administración de inventario	61
Figura 6.5. Diagrama de caso de uso: Administración de ventas.	65
Figura 6.6. Página para Iniciar sesión	71
Figura 6.7. Formulario de Movimientos de la kardex.....	71
Figura 6.8. Formulario para Registrar movimiento de la kardex.....	72
Figura 6.9. Formulario para Facturación.....	72
Figura 6.10. Formulario para Devoluciones	73
Figura 6.11. Formulario para Registro de retenciones.....	73
Figura 6.12. Formulario para Generar proformas	74
Figura 6.13. Diagrama de clases del módulo de gestión de inventario y facturación	75
Figura 6.14. Diagrama de secuencia: Interacción entre módulos	76
Figura 6.15. Diagrama de secuencia: Iniciar sesión.....	76
Figura 6.16. Diagrama de secuencia: Registrar movimientos en kardex	77
Figura 6.17. Diagrama de secuencia: Revisar movimientos del kardex	77
Figura 6.18. Diagrama de secuencia: Toma de inventario.....	78
Figura 6.19. Diagrama de secuencia: Ajuste de inventario.....	78
Figura 6.20. Diagrama de secuencia: Consultar stock y costos de mercadería.....	79
Figura 6.21. Diagrama de secuencia: Facturación	79
Figura 6.22. Diagrama de secuencia: Devoluciones	80
Figura 6.23. Diagrama de secuencia: Registro de retenciones.....	80
Figura 6.24. Diagrama de secuencia: Administración de formas de pago.....	81
Figura 6.25. Diagrama de actividades: Administración de inventario.....	81
Figura 6.26. Diagrama de actividades: Facturación y devoluciones.....	82
Figura 6.27. Diagrama de actividades: Registro de retenciones	82
Figura 6.28. Diagrama de actividades: Proformas y formas de pago	83
Figura 6.29. Diagrama de despliegue: Módulo de gestión de inventario y facturación.....	83

Figura 6.30. Niveles y capas de la SOA	87
Figura 6.31. Diseño de base de datos Módulo de gestión de inventario y facturación	88
Figura 6.32. Código de prueba - Método insert	127
Figura 6.33. Código de prueba - Clase Bsx_retención_ventaTest	127
Figura 6.34. JUnit – Reporte de fallos	128
Figura 6.35. JUnit – Reporte de la prueba unitaria	128

ÍNDICE DE TABLAS

CAPÍTULO III METODOLOGÍA

Tabla 3.1. Operacionalización de variables – variable independiente	31
Tabla 3.2. Operacionalización de variables – variable dependiente	32

CAPÍTULO IV ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

Tabla 4.1. Cuadro de Frecuencia: Pregunta 1	34
Tabla 4.2. Cuadro de Frecuencia: Pregunta 2	35
Tabla 4.3. Cuadro de Frecuencia: Pregunta 4	36
Tabla 4.4. Cuadro de Frecuencia: Pregunta 4	37
Tabla 4.5. Cuadro de Frecuencia: Pregunta 5	38
Tabla 4.6. Cuadro de Frecuencia: Pregunta 6	39
Tabla 4.7. Cuadro de Frecuencia: Pregunta 7	40
Tabla 4.8. Cuadro de Frecuencia: Pregunta 8	41
Tabla 4.9. Cuadro de Frecuencia: Pregunta 9	42
Tabla 4.10. Cuadro de Frecuencia: Pregunta 10	43

CAPÍTULO VI PROPUESTA

Especificaciones de casos de uso

Tabla 6.1. Caso de uso: Iniciar Sesión	61
Tabla 6.2. Caso de uso: Registrar movimientos en kardex	62
Tabla 6.3. Caso de uso: Revisar movimientos del kardex	62
Tabla 6.4. Caso de uso: Toma de inventario	63

Tabla 6.5. Caso de uso: Ajuste de inventario.....	64
Tabla 6.6. Caso de uso: Consultar stock y costos de mercadería.....	64
Tabla 6.7. Caso de uso: Facturación	65
Tabla 6.8. Caso de uso: Devoluciones	66
Tabla 6.9. Caso de uso: Registro de retenciones.....	67
Tabla 6.10. Caso de uso: Proformas.....	68
Tabla 6.11. Caso de uso: Administración de formas de pago.....	69
Tabla 6.12. Caso de uso: Administración de reportes.....	70

Diccionario de datos del Módulo de Gestión de Inventario y Facturación.

Tabla 6.13. Descripción de la tabla BSX_BODEGA	89
Tabla 6.14. Descripción de la tabla BSX_AGENCIA_BODEGA	89
Tabla 6.15. Descripción de la tabla BSX_TARJETA_KARDEX	90
Tabla 6.16. Descripción de la tabla BSX_TKDETALLE.....	91
Tabla 6.17. Descripción de la tabla BSX_SERIALES	93
Tabla 6.18. Descripción de la tabla BSX_TIPO_FORMA_PAGO	94
Tabla 6.19. Descripción de la tabla BSX_VENTA.....	94
Tabla 6.20. Descripción de la tabla BSX_VENTA_DETALLE.....	96
Tabla 6.21. Descripción de la tabla BSX_FORMA_PAGO_VENTA	97
Tabla 6.22. Descripción de la tabla BSX_RETENCION_VENTA.....	97
Tabla 6.23. Descripción de la tabla BSX_RETENCION_VENTA_DETALLE	98
Tabla 6.24. Descripción de la tabla BSX_PROFORMA	99
Tabla 6.25. Descripción de la tabla BSX_PROFORMA_DETALLE	101
Tabla 6.26. Descripción de la tabla BSX_NOTA	102
Tabla 6.27 Descripción de la tabla BSX_DETALLE_NOTA.....	103

RESUMEN EJECUTIVO

En este documento se presentará la propuesta de elaboración de una aplicación web para la GESTIÓN DE INVENTARIO Y FACTURACIÓN de la Empresa BESIXPLUS CIA LTDA. El Trabajo Estructurado de Manera Independiente (TEMI) está separado en 6 capítulos donde se tratarán diversos aspectos de la aplicación como las bases sobre la cual parte la propuesta realizada, su diseño, desarrollo y los resultados obtenidos durante las diferentes pruebas realizadas antes y después de su implantación.

En el **CAPÍTULO I** de la documentación de este proyecto se hablará sobre el impacto que tiene a nivel meso, maso y micro sobre la no automatización de los procesos de gestión de inventario y facturación. En este capítulo plantaremos el problema y los objetivos que el TEMI resolverá.

En el **CAPÍTULO II** se expondrá el marco teórico del tema de investigación, antecedentes investigativos del TEMI y fundamentación legal de la Empresa BESIXPLUS CIA. LTDA. donde se implantará la aplicación.

En el **CAPÍTULO III** se explicará la metodología de investigación, factibilidad del proyecto y el plan de recolección de la información.

En el **CAPÍTULO IV** se realizará el análisis e interpretación de los resultados de las entrevistas realizadas.

En el **CAPÍTULO V** se incluirán las conclusiones y recomendaciones de la investigación.

En el **CAPÍTULO VI** se expondrá la propuesta de la investigación objetivos, factibilidad técnica y económica, metodología, diagramas UML, pruebas de sistema, conclusiones y recomendaciones de la implantación.

INTRODUCCIÓN

BESIXPLUS CIA LTDA. una empresa dedicada al comercio de toda clase de productos y servicios computacionales e informáticos, desarrollo y distribución de software, ha optado por implantar un sistema para el manejo de inventario y facturación acorde a las necesidades de su lógica de negocio.

En el país existen software desarrollados con propósitos similares a los que maneja BESIXPLUS CIA LTDA., pero ninguno ha logrado satisfacer las expectativas de la empresa por lo que se será necesario crear un sistema que cumpla todos los requerimientos y de solución a los problemas de la empresa.

La evolución de los lenguajes de programación y arquitecturas de software es constante lo cual conlleva a que el sistema a desarrollarse se base en tecnologías actuales.

La internet como medio de comunicación masivo orienta a que la mayor parte de aplicaciones se desarrollen en un ambiente web, siendo una de las principales ventajas la accesibilidad y multiplataforma de la aplicación, permitiendo a los usuarios del sistema acceder al mismo desde cualquier lugar del mundo sin importar la plataforma operativa que este corriendo.

Mediante este proyecto se pretende dar solución a los procesos administrativos que conlleva la gestión de inventario y facturación, su integración al Sistema Administrativos Contable permitirá crear una herramienta sólida que apoye a las actividades de la empresa.

CAPÍTULO I

EL PROBLEMA DE INVESTIGACIÓN

1.1. TEMA

“Módulo de gestión de inventario y facturación bajo arquitectura SOA para el Sistema Administrativo Contable de la Empresa BESIXPLUS CIA. LTDA.”

1.2. PLANTEAMIENTO DEL PROBLEMA

1.2.1. Contextualización

El manejo manual de los procesos administrativos de inventario y facturación es un problema considerable en el Ecuador, esto ha provocado que las empresas brinden un servicio deficiente y de poca calidad, la descoordinación que existe entre empleados al momento de procesar y almacenar los datos conlleva a que estos sean guardados incompletos y en muchos casos duplicados, ocasionando pérdidas económicas y poca competitividad en su medio.

En la provincia de Tungurahua las empresas han incrementado su producción y por ende los procesos administrativos son cada vez más complejos de manejarlos, el desempeño de las empresas baja notablemente provocando la pérdida de clientes lo cual conlleva a crisis económicas empresariales.

En la empresa BESIXPLUS CIA LTDA. de la ciudad de Ambato se ha podido observar que los procesos de inventario y facturación se realizan manualmente

debido a la ausencia de software que los maneje de forma automatizada, ocasionando en la empresa inconsistencias en los estados financieros y la poca o nula incursión en mercados internacionales.

1.2.2. Análisis crítico

La Empresa BESIXPLUS CIA LTDA. posee un sistema manual para el proceso y almacenamiento de las transacciones de gestión de inventario y facturación, haciendo que en muchas ocasiones la información sea guardada incompleta, con errores y en algunos casos duplicada, motivo por el cual el servicio a clientes y proveedores es lento.

La falta de personal en la empresa ha provocado que el inventario no se lleve de manera ordenada, el listado de los productos existentes en bodega no está actualizado, debido a esto atender a los clientes con sus pedidos resulta en algunas ocasiones imposible, las ventas bajan considerablemente dando gran oportunidad a la competencia.

El almacenamiento de los documentos se lo hace en un archivero, aquí se colocan carpetas con facturas, órdenes de compra, ordenes de pagos, inventarios, nóminas y más, no existe seguridad alguna haciendo que el robo de información sea insospechable.

Los empleados de BESIXPLUS CIA LTDA. tienen poco conocimiento en el área de contabilidad no están capacitados para llevar a cabo los procesos contables que genera la gestión de inventario y facturación, en muchas ocasiones no se ha podido contar con un estado real de la situación económica por lo tanto es imposible tomar decisiones a futuro provocando que la empresa tenga poca incursión en el mercado.

1.2.3. Prognosis

Si la Empresa BESIXPLUS CIA LTDA. no integra el módulo de gestión de inventario y facturación al Sistema Administrativo Contable puede llegar a originar:

- Una considerable pérdida económica.
- Disminución de las ventas en la zona centro del país.
- La imagen de la empresa se deteriore debido a que el cliente insatisfecho puede hablar mal de la misma.
- La no incursión en mercados internacionales.

1.3. FORMULACIÓN DEL PROBLEMA

¿Cómo incide el desarrollo e integración del módulo de gestión de inventario y facturación bajo arquitectura SOA (Arquitectura orientada a servicios) al Sistema Administrativo Contable de la Empresa BESIXPLUS CIA LTDA.?

1.3.1. Preguntas directrices

- ¿Qué es la arquitectura SOA?
- ¿Qué conceptos de SOA se aplicará para el desarrollo del módulo?
- ¿Qué lenguaje de programación se utilizará para aplicar la arquitectura SOA en el desarrollo del módulo?
- ¿Qué pasos hay que seguir para aplicar la arquitectura SOA en el desarrollo del módulo?
- ¿Qué motor de base de datos será el más adecuado de utilizar para el desarrollo del módulo?
- ¿Cómo maneja los procesos de inventario y facturación?
- ¿Cuándo se acepta la devolución de un producto?
- ¿Cómo se controla el stock de un producto ?
- ¿Proporcionará mejor organización en las tareas diarias de la empresa?
- ¿Se necesitará capacitar al personal de la empresa?

1.3.2. Delimitación del objeto de investigación

Delimitación de contenido

- Campo: Administrativo – Informático
- Área : Administración y contabilidad
- Aspecto: Sistema de gestión de Inventarios y facturación

Delimitación temporal

- El tiempo de ejecución de la investigación abarcará el período noviembre 2010 - noviembre del 2011.

Delimitación espacial

- El siguiente proyecto se desarrollará en la Empresa BESIXPLUS CIA. LTDA. ubicada en la calle Seymour 0-13 entre Tortuga y Camino del Rey de la ciudad de Ambato provincia de Tungurahua.

1.4. JUSTIFICACIÓN

El proyecto a realizarse está basado en conocimientos previos adquiridos en la empresa BESIXPLUS CIA LTDA. el propósito del mismo es automatizar el proceso de gestión de inventario y facturación para poder agilizar la emisión de facturas, tener un control actualizado de productos en bodega, y así brindar un mejor servicio al cliente.

La importancia del presente proyecto se basa en la utilización de software libre para el desarrollo de la aplicación, bajo licencias GPL (Licencia Pública General de GNU) en lenguajes de programación, servidores de aplicaciones, servidor de reportes y licencia BSD para base de datos.

También cabe destacar que el proyecto se desarrollará 100% en ambiente web utilizando arquitectura SOA, teniendo como unas de las principales ventajas la accesibilidad y multiplataforma, permitiendo a los usuarios del sistema acceder al

mismo desde cualquier lugar del mundo sin importar la plataforma operativa que este corriendo.

Este proyecto es factible y viable ya que dará solución a los problemas actuales que tiene la empresa con la gestión de inventario y facturación en el Sistema Administrativo Contable.

1.5. OBJETIVOS

1.5.1. Objetivo General

Diseñar un módulo de gestión de inventario y facturación bajo arquitectura SOA que permita optimizar el Sistema Administrativo Contable de la Empresa BESIXPLUS CIA. LTDA.

1.5.2. Objetivos Específicos

- Analizar los procesos manuales de inventario y facturación que se realiza para el registro, eliminación y consulta de productos.
- Utilizar la arquitectura de software SOA en el desarrollo de la aplicación.
- Proveer información a los diferentes módulos que forman el Sistema Administrativo Contable de la Empresa.
- Desarrollar e integrar el módulo de gestión de inventarios y facturación bajo arquitectura SOA al Sistema Administrativo Contable de la Empresa.

CAPÍTULO II

MARCO TEÓRICO

2.1. ANTECEDENTES INVESTIGATIVOS

En la Empresa BESIXPLUS CIA LTDA. no se ha desarrollado o adquirido ningún software que maneje el inventario y facturación, para el proceso de los mismos utilizan herramientas basadas en hojas de cálculo computarizadas y la ventas se hace manualmente utilizando un facturero.

Una vez revisado las tesis y pasantías de la Facultad de Ingeniería en Sistemas Electrónica e Industrial de la Universidad Técnica de Ambato, se ha recolectado la siguiente información:

Los estudiantes Guzmán Meléndez Brigida Gissela y Tabares Burbano Cristian Vinicio realizaron el tema: SISTEMA PARA EL CONTROL DE INVENTARIOS Y FACTURACIÓN EN UNA DISTRIBUIDORA FARMACÉUTICA.

Los estudiantes Núñez Rodríguez Flor María y Vásquez Freire Edison Fabián realizaron el tema: SISTEMA CONTROL DE FACTURACIÓN CRÉDITOS Y BODEGA BAJO ARQUITECTURA CLIENTE SERVIDOR EN LA EMPRESA MERCANTIL GARZOZI SUCURSAL AMBATO.

El estudiante Freire Freire Hugo Leonidas realizo el tema: DESARROLLO E IMPLANTACIÓN DE UN SISTEMA PARA EL CONTROL Y FACTURACIÓN DEL AGUA POTABLE EN EL MUNICIPIO DE TISALEO.

2.2. FUNDAMENTACIÓN LEGAL

2.2.1. Empresa BESIXPLUS CIA. LTDA.

El desarrollo del módulo de gestión de inventario y facturación, está relacionado con la Empresa BESIXPLUS CIA. LTDA. una compañía joven y emprendedora que nace con el fin de brindar servicios informáticos especializados y de calidad, a todo tipo de empresas con soporte profesional en lo que concierne a código abierto y propietario de software.

La empresa cuenta con los siguientes estatutos: **CAPITULO PRIMERO.-** de la compañía, su denominación, objeto social, duración, domicilio, nacionalidad, transformación, fusión y escisión.- **Artículo primero.- Denominación:** La compañía se denominará **BESIXPLUS CIA. LTDA.- Artículo Segundo.- Objeto Social:** La compañía tendrá por objeto social: **a)** La importación y exportación, de toda clase de productos bienes y servicios computacionales e informáticos; **b)** Desarrollo y comercialización de productos de software en representación de compañías extranjeras en el Ecuador. **c)** La compra, venta y distribución de equipos de cómputo. **d)** Diseño, construcción y mantenimiento de redes de transmisión de información **e)** Compra venta de dominios y hosting **f)** La compañía no podrá constituirse en garante o fiadora de obligaciones distintas de las suyas propias sin que medie autorización escrita previa y expresa de la Junta General de Socios. Para el cumplimiento de su objeto, la Compañía podrá intervenir como socio o accionista en la formación de toda clase de sociedades o compañías, aportar capital a las mismas o adquirir, tener y poseer acciones, obligaciones o participaciones en otras compañías. En general, la Compañía podrá realizar toda clase de actos, contratos y operaciones permitidas por las leyes ecuatorianas, que sean acordes con su objeto y necesarias y convenientes para su cumplimiento.

La empresa BESIXPLUS CIA LTDA. no cuenta con estatutos legales que respalden sus procesos de inventario y facturación.

Ley Orgánica de Régimen Tributario Interno

LEY DE RÉGIMEN TRIBUTARIO INTERNO

(Codificación No. 2004-026)

Título Segundo

IMPUESTO AL VALOR AGREGADO

Capítulo I

OBJETO DEL IMPUESTO

Art. 52.- Objeto del impuesto.- Establécese el impuesto al valor agregado (IVA), que grava al valor de la transferencia de dominio o a la importación de bienes muebles de naturaleza corporal, en todas sus etapas de comercialización, y al valor de los servicios prestados, en la forma y en las condiciones que prevé esta Ley.

Capítulo II

HECHO IMPONIBLE Y SUJETOS DEL IMPUESTO

Art. 61.- Hecho generador.- (Reformado por el Art. 117 de la Ley s/n, R.O. 242-3S, 29-XII-2007).- El IVA se causa en el momento en que se realiza el acto o se celebra el contrato que tenga por objeto transferir el dominio de los bienes o la prestación de los servicios, hecho por el cual se debe emitir obligatoriamente la respectiva factura, nota o boleta de venta.

En el caso de los contratos en que se realice la transferencia de bienes o la prestación de servicios por etapas, avance de obras o trabajos y en general aquellos que adopten la forma de tracto sucesivo, el impuesto al valor agregado –

IVA se causará al cumplirse las condiciones para cada período, fase o etapa, momento en el cual debe emitirse el correspondiente comprobante de venta.

En el caso de introducción de mercaderías al territorio nacional, el impuesto se causa en el momento de su despacho por la aduana.

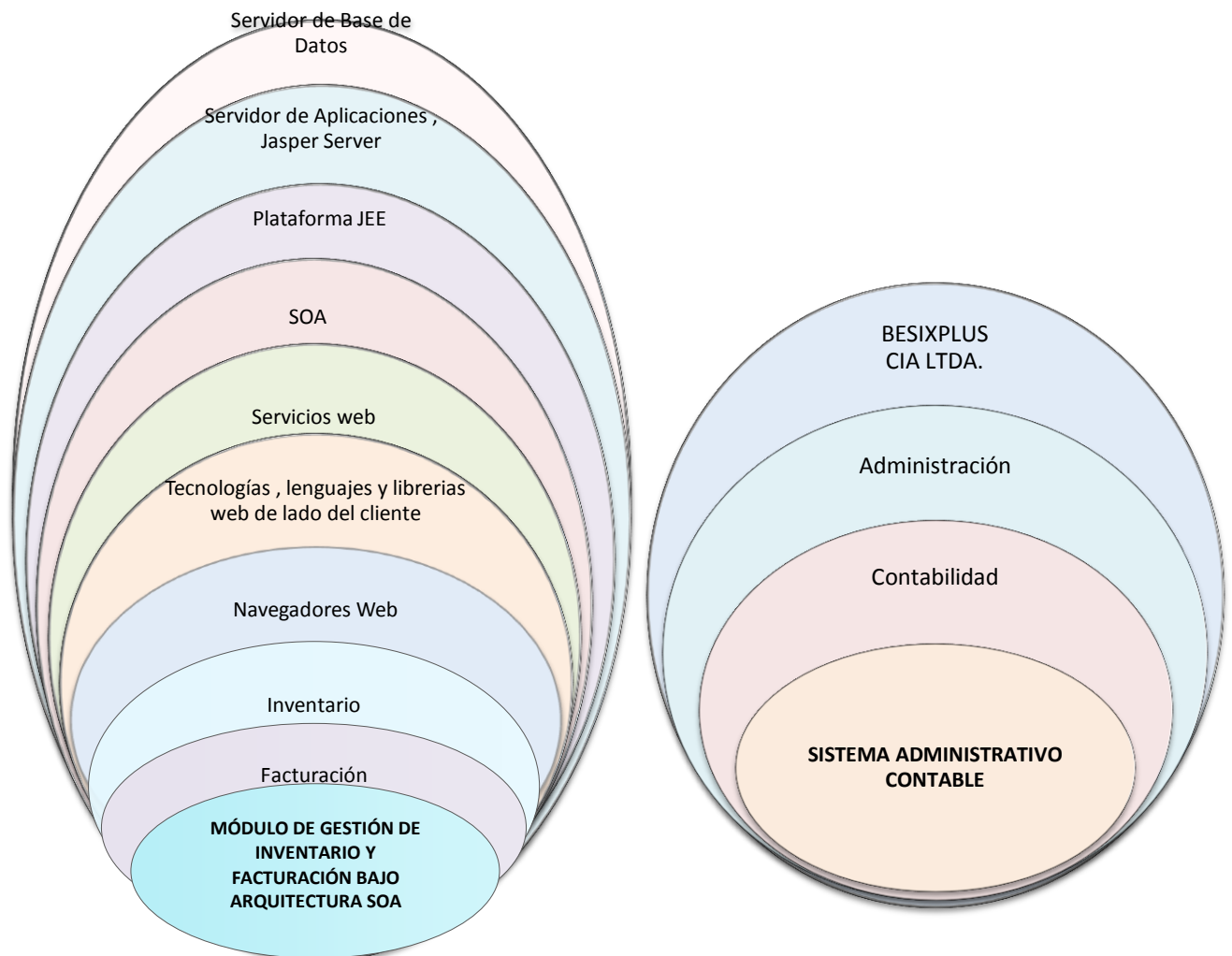
Capítulo III

TARIFA DEL IMPUESTO Y CRÉDITO TRIBUTARIO

Art. 65.- Tarifa.- La tarifa del impuesto al valor agregado es del 12%.

2.3. CATEGORÍAS FUNDAMENTALES

Figura 2.1. Categorías fundamentales



Arquitectura del Software

En el sitio Web de **Josep, Casanova** dice: “La Arquitectura del Software es el diseño de más alto nivel de la estructura de un sistema, programa o aplicación y tiene la responsabilidad de:

- Definir los módulos principales
- Definir las responsabilidades que tendrá cada uno de estos módulos
- Definir la interacción que existirá entre dichos módulos.
- Control y flujo de datos
- Secuencia de la información
- Protocolos de interacción y comunicación
- Ubicación en el hardware “

(http://www.alzado.org/articulo.php?id_art=355)

Existen diferentes tipos de arquitectura del software entre estas tenemos:

- **Monolítica**

En el sitio Web de **Wikipedia** dice: “Donde el software se estructura en grupos funcionales muy acoplados. Están constituidos fundamentalmente por un solo programa compuesto de un conjunto de rutinas entrelazadas de tal forma que cada una puede llamar a cualquier otra. Generalmente están hechos a medida, por lo que son eficientes y rápidos en su ejecución y gestión, pero por lo mismo carecen de flexibilidad para soportar diferentes ambientes de trabajo o tipos de aplicaciones.” ([http://es.wikipedia.org/w/index.php?title=Arquitectura_monol%C3%ADtica_\(inform%C3%A1tica\)&action=edit&redlink=1](http://es.wikipedia.org/w/index.php?title=Arquitectura_monol%C3%ADtica_(inform%C3%A1tica)&action=edit&redlink=1))

- **Cliente-servidor**

En el sitio Web de **Wikipedia** dice: “Es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, que le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.” (<http://es.wikipedia.org/wiki/Cliente-servidor>)

- **Arquitectura de tres niveles**

En el sitio Web de **Wikipedia** dice: “Especialización de la arquitectura cliente-servidor donde la carga se divide en tres partes (o capas) con un reparto claro de funciones: una capa para la presentación (interfaz de usuario), otra para el cálculo (donde se encuentra modelado el negocio) y otra para el almacenamiento (persistencia).”

(http://es.wikipedia.org/wiki/Arquitectura_de_software)

- **SOA**

En el sitio Web de **Wikipedia** dice: “Es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requisitos del negocio.” (Ver en 2.3.5).

(http://es.wikipedia.org/wiki/Arquitectura_orientada_a_servicios)

2.3.1. Servidor de base de datos

Un servidor de base de datos es un programa que provee servicios de base de datos a otros programas u otras computadoras, como es definido por el modelo cliente-servidor. Base de datos se define como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa. Los SGBD (Sistemas de Gestión de base de datos) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje

de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Entre algunos de los sistemas de gestión de base de datos libres tenemos: PostgreSQL y MySQL y con licencias Oracle y SQL Server como los más conocidos.

2.3.1.1. PostgreSQL

Figura 2.2. Logo de PostgreSQL



En el sitio Web de **PostgreSQL** dice: “Es un sistema de base de datos objeto-relacional de gestión (ORDBMS), basado en POSTGRES, Versión 4.2, desarrollado en la Universidad de California en Berkeley del Departamento de Ciencias de la Computación. Postgres fue pionera en muchos conceptos que sólo estuvo disponible en algunos sistemas de bases de datos comerciales mucho más tarde.

PostgreSQL es un descendiente de código abierto del código original de Berkeley. Es compatible con una gran parte del estándar SQL y ofrece muchas características modernas:

- Consultas complejas
- Claves externas
- Disparadores
- Puntos de vista
- Integridad de las transacciones
- Multiversión control de concurrencia

Además, PostgreSQL puede ser ampliado por el usuario en muchos aspectos, por ejemplo mediante la adición de nuevos

- Los tipos de datos
- Funciones
- Los operadores
- Funciones de agregación
- Métodos de índice
- Lenguajes de procedimiento

Y debido a la licencia liberal, PostgreSQL puede ser usado, modificado y distribuido por todo el mundo de forma gratuita para cualquier propósito, ya sea privado, comercial o académico.

PostgreSQL cuenta con características avanzadas tales como:

- Multi-Version Control de concurrencia (MVCC)
- Punto en el tiempo de recuperación
- Tablespaces
- Replicación asincrónica
- Backups en caliente,
- Tiene soporte completo para subconsultas (incluyendo subconsultas en la cláusula FROM).
- Es compatible con conjuntos de caracteres internacionales, las codificaciones de caracteres multibyte, Unicode.
- Cumple con la norma estándar ANSI-SQL 2008.

- Es altamente escalable, tanto en la gran cantidad de datos que puede manejar y en el número de usuarios concurrentes que puede soportar.

Algunos de los límites de PostgreSQL.

Límite	Valor
Tamaño máximo de base de datos	Ilimitado
Tamaño máximo de tablas	32 TB
Tamaño máximo de filas	1.6 TB
Tamaño máximo de campo	1 GB
Máximo de filas por tabla	Ilimitado
Máximo de columnas por tabla	250 - 1600 depende del tipo de columna
Máximo de índices por tabla	Ilimitado

(<http://www.postgresql.org/docs/8.1/static/preface.html#INTRO-WHATIS>)

(<http://www.postgresql.org/about/>)

2.3.2. Servidor de aplicaciones

En el sitio Web de **Editum.org** dice: “Un servidor de aplicaciones es un software que proporciona aplicaciones a los equipos o dispositivos cliente, por lo general a través de Internet y utilizando el protocolo http. Los servidores de aplicación se distinguen de los servidores Web por el uso extensivo del contenido dinámico y por su frecuente integración con bases de datos.

Además, Un servidor de aplicaciones es un producto basado en un componente que se encuentra en el plano medio de la arquitectura central de un servidor. Proporciona servicios de ‘middleware’, es decir, trabaja como un intermediario para la seguridad y el mantenimiento, además de proveer acceso a los datos.”

(<http://www.editum.org/Servidor-De-Aplicaciones-Que-Es-p-604.html>)

Entre algunos de los servidores de aplicaciones tenemos: JBoss, IIS, IBM WebSphere, y WebLogic de BEA.

2.3.2.1. JBoss

Figura 2.3. Logo de JBoss



En el sitio Web de **Scribd** dice: “JBoss es un servidor de aplicaciones J2EE de código abierto implementado en Java puro. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo que lo soporte.

Características:

- Producto de licencia de código abierto sin coste adicional.
- Cumple los estándares.
- Confiable a nivel de empresa
- Incrustable, orientado a arquitectura de servicios.
- Flexibilidad consistente
- Servicios del middleware para cualquier objeto de Java”

(<http://es.scribd.com/doc/58116358/Jboss>)

2.3.3. Jasper Server

En el sitio Web de **ptb** dice: “Es el contenedor de reportes en un ambiente de acceso seguro a los usuarios y recursos. Puede ser usado stand-alone o desde otras aplicaciones. Se puede personalizar, ejecutar, programar y distribuir los reportes a usuarios y grupos. Administra los recursos compartidos (imágenes, datos, etc).”
(http://www.ptb.com.mx/index.php?option=com_content&view=article&id=44&Itemid=54)

- **Jasper Report**

En el sitio Web de **CEPEU** dice: “Es una potente herramienta Open Source para generar reportes en Java, con la habilidad de producir contenido completo para la pantalla, directo para impresora o en diferentes formatos de archivos(PDF, HTML, XLS, CSV y XML, entre otros).

La librería es 100% Java, y puede reutilizarse tanto en plicaciones cliente y cliente/servidor, como en aplicaciones web, J2EE, etc. JasperReports permite organizar la información obtenida desde una base de datos relacional, a través de conectores JDBC, en diseños de reportes predefinidos en un formato XML. (http://www.cepeu.edu.py/LIBROS_ELECTRONICOS_3/lpcu089%20-%2001.pdf)

- **i Report**

En el sitio Web de **Cristhian Herrera** dice: “La herramienta iReport es un constructor / diseñador de informes visual, poderoso, intuitivo y fácil de usar para JasperReports escrito en Java. Este instrumento permite que los usuarios corrijan visualmente informes complejos con cartas, imágenes, subinformes, etc. iReport está además integrado con JFreeChart, una de la biblioteca gráficas OpenSource más difundida para Java. Los datos para imprimir pueden ser recuperados por varios caminos incluso múltiples uniones JDBC, TableModels, JavaBeans, XML, etc.”

(<http://www.adictosaltrabajo.com//tutoriales/tutoriales.php?pagina=ireport>)

2.3.4. Plataforma JEE (Java Enterprise Edition)

En el documento Web de Evelio Clavel Rosales y Dariel Membribes dice: “JEE es una plataforma que provee una arquitectura robusta para el desarrollo de aplicaciones empresariales en el lenguaje Java utilizando un modelo multicapas. Incluye una serie de APIs (Application Programming Interface), tecnologías, herramientas de desarrollo y especificaciones e implementaciones de referencia de los servicios que brinda la plataforma. Facilita el desarrollo de aplicaciones portables, escalables y seguras ya que incluye soluciones para cuestiones comunes

como transacciones, distribución de objetos remotos, soporte de servicios Web, trabajo con XML entre otras.” (<http://semanatecnologica.fordes.co.cu/ocs-2.3.2/public/site/165.pdf>)

2.3.4.1. Herramientas de desarrollo JEE

Eclipse

Eclipse es un Entorno Integrado de Desarrollo IDE (Integrated Development Environment), para todo tipo de aplicaciones libres, inicialmente desarrollado por IBM, y actualmente gestionado por la Fundación Eclipse.

Herramienta para el programador desarrollada principalmente para el desarrollo de aplicaciones Java, facilitando al máximo la gestión de proyectos colaborativos mediante el control de versiones 'cvs', es posible también con subversión, exportar e importar proyectos. Permite añadir nuevas funcionalidades al editor, a través de nuevos módulos ('plugins'), para programar en otros lenguajes de programación además de Java como C/C++, PHP, Python, Ruby, Cobol y más.

Características

- Multiplataforma (GNU/Linux, Solaris, Mac OSX, Windows)
- Estructura de plug-in que hace sencillo añadir nuevas características y funcionalidades.
- Asistentes (wizards) para la creación, exportación e importación de proyectos; para generar esqueletos de códigos (templates).

(<https://forja.rediris.es/docman/view.php/970/1431/memoria.pdf>)

2.3.4.2. Lenguaje de programación Java

En el documento Web de Francisco Javier Pavón Molina dice: “Java es un lenguaje de desarrollo de propósito general, y como tal es válido para realizar todo tipo de aplicaciones profesionales.

Características de Java

- Es intrínsecamente orientado a objetos.
- Funciona perfectamente en red.
- Aprovecha características de la mayoría de los lenguajes modernos evitando sus inconvenientes. En particular los del C++.
- Tiene una gran funcionalidad gracias a sus librerías (clases).
- No tiene punteros manejables por el programador, aunque los maneja interna y transparentemente.
- El manejo de la memoria no es un problema, la gestiona el propio lenguaje y no el programador.
- Genera aplicaciones con pocos errores posibles.
- Incorpora Multi-Threading (para permitir la ejecución de tareas concurrentes dentro de un mismo programa).

El lenguaje Java es robusto. Las aplicaciones creadas en este lenguaje son susceptibles de contener pocos errores, principalmente porque la gestión de memoria y punteros es realizada por el propio lenguaje y no por el programador. Bien es sabido que la mayoría de los errores en las aplicaciones vienen producidos por fallos en la gestión de punteros o la asignación y liberación de memoria. Además, el lenguaje contiene estructuras para la detección de excepciones (errores de ejecución previstos) y permite obligar al programador a escribir código fiable mediante la declaración de excepciones posibles para una determinada clase reutilizable.”

(<http://www.aiu.edu/applications/DocumentLibraryManager/upload/Principles%20of%20Programming%20II.pdf>)

2.3.5. SOA

En el sitio Web de **IBM** dice: “La arquitectura SOA (Arquitectura Orientada a Servicios) consiste en un método de diseño de software donde las aplicaciones de negocio se descomponen en “servicios” individuales que pueden ser utilizados independientemente de las aplicaciones de las que forman parte y de las plataformas informáticas sobre las que se ejecutan. Al poder disponer de los servicios individuales de las aplicaciones como piezas independientes, las empresas tendrán la posibilidad de integrarlos y agruparlos de maneras distintas para conseguir capacidades completamente nuevas.” (<http://www-304.ibm.com/easyaccess/files/serve?contentid=129573>)

Principios de SOA

En el sitio Web de **Trumbulianos** dice:

- “Los Servicios deben ser reusables: Todo servicio debe ser diseñado y construido pensando en su reutilización dentro de la misma aplicación, dentro del dominio de aplicaciones de la empresa o incluso dentro del dominio público para su uso masivo.
- Los Servicios deben proporcionar un contrato formal: Todo servicio desarrollado, debe proporcionar un contrato en el cual figuren: el nombre del servicio, su forma de acceso, las funcionales que ofrece, los datos de entrada de cada una de las funcionalidades y los datos de salida. De esta manera, todo consumidor del servicio, accederá a este mediante el contrato, logrando así la independencia entre el consumidor y la implementación del propio servicio.
- Los Servicios deben tener bajo acoplamiento: Es decir, que los servicios tienen que ser independientes los unos de los otros. Para lograr ese bajo acoplamiento, lo que se hará es que cada vez que se vaya a ejecutar un servicio, se accederá a él a través del contrato, logrando así la

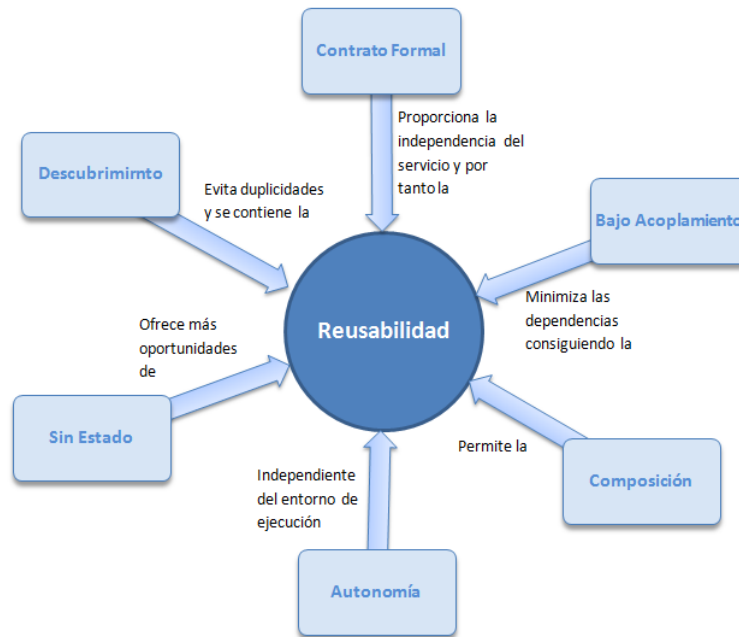
independencia entre el servicio que se va a ejecutar y el que lo llama. De esta manera serán totalmente reutilizables.

- Los Servicios deben permitir la composición: Todo servicio debe ser construido de tal manera que pueda ser utilizado para construir servicios genéricos de más alto nivel, el cual estará compuesto de servicios de más bajo nivel. En el caso de los Servicios Web, esto se logrará mediante el uso de los protocolos para orquestación (WS-BPEL) y coreografía (WS-CDL).
- Los Servicios deben de ser autónomos: Todo Servicio debe tener su propio entorno de ejecución. De esta manera el servicio es totalmente independiente y nos podemos asegurar que así podrá ser reutilizable desde el punto de vista de la plataforma de ejecución.
- Los Servicios no deben tener estado: Un servicio no debe guardar ningún tipo de información. Esto es así porque una aplicación está formada por un conjunto de servicios, lo que implica que si un servicio almacena algún tipo de información, se pueden producir problemas de inconsistencia de datos. La solución, es que un servicio sólo contenga lógica, y que toda información esté almacenada en algún sistema de información sea del tipo que sea.
- Los Servicios deben poder ser descubiertos: Todo servicio debe poder ser descubierto de alguna forma para que pueda ser utilizado, consiguiendo así evitar la creación accidental de servicios que proporcionen las mismas funcionalidades. En el caso de los Servicios Web, el descubrimiento se logrará publicando los interfaces de los servicios en registros UDDI (Universal Discovery Description and Integration). “

(<http://arquiware.blogspot.com/>)

Interrelación entre los principios de SOA. (Ver Figura 2.4)

Figura 2.4. Interrelación entre los principios de SOA.



Obtenido de: <http://arquiware.blogspot.com/>

2.3.6. Servicios web

En el sitio Web de W3C dice: “Los servicios Web son aplicaciones modulares auto-descriptivas que se pueden publicar, ubicar e invocar desde cualquier punto de la Web o desde el interior de una red local basada en estándares abiertos de Internet.

Los servicios Web son construidos alrededor de un conjunto de protocolos tales como XML (Extensible Markup Lenguaje), SOAP (Simple Object Access Protocol) y WSDL (Web Service Description Language Document).

Son tres los componentes principales en los servicios Web, el registro de servicios, el proveedor de los servicios y el consumidor o cliente de los servicios.

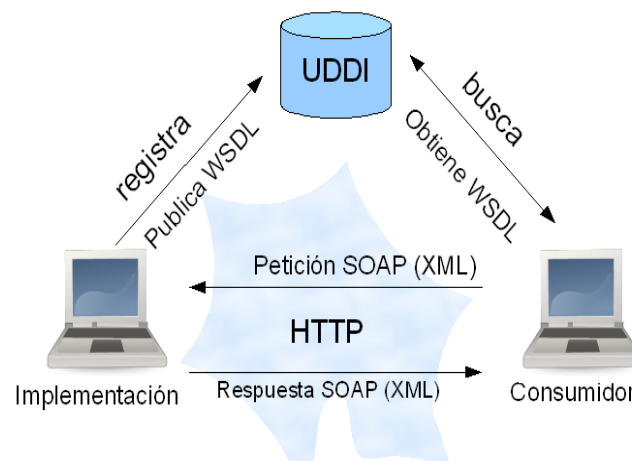
Un registro es un catálogo para buscar en línea servicios y es utilizado por un proveedor para publicar sus servicios.

El proveedor de servicios es una entidad u organización dispuesta a proporcionar datos o servicios. Un proveedor de servicios crea la descripción del servicio usando WSDL que contiene una lista de operaciones espaciales y no espaciales soportadas, tipos de datos intercambiados y la localización del servicio.

El cliente de los servicios puede ser una organización o persona con necesidad de datos o servicios.

En los servicios Web se utiliza SOAP para el intercambio de datos entre el proveedor de los servicios y los usuarios o clientes. SOAP es un protocolo ligero para el intercambio de información en un entorno descentralizado y distribuido.” (<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>)

Figura 2.5. Proceso de consumo de un Servicio Web.



Obtenido de: <http://agronomiaunlu.blogspot.com/>

2.3.7. Tecnologías, lenguajes y librerías web de lado del cliente

2.3.7.1. AJAX (Asynchronous Javascript y XML)

En el sitio Web de **Mayer Horna García y Jonathan Lara Vazquez** dice:”Es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications), implica el uso coordinado de distintas tecnologías que en conjunto permiten una mayor rapidez y eficacia para las aplicaciones basadas en la web. Estas aplicaciones se ejecutan en el navegador de los usuarios mientras se

mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la velocidad y usabilidad en las aplicaciones.” (<http://www.slideshare.net/njelov/ajax-huancayo>)

2.3.7.2. HTML

En el sitio Web de **Wikipedia** dice: “HTML, siglas de *HyperText Markup Language* («lenguaje de marcado de hipertexto»), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<, >). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un *script* (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.” (<http://es.wikipedia.org/wiki/HTML>)

2.3.7.3. DHTML

En el sitio Web de **Wikipedia** dice: “El HTML Dinámico o DHTML (del inglés *Dynamic HTML*) designa el conjunto de técnicas que permiten crear sitios web interactivos utilizando una combinación de lenguaje HTML estático, un lenguaje interpretado en el lado del cliente (como JavaScript), el lenguaje de hojas de estilo en cascada (CSS) y la jerarquía de objetos de un DOM.” (http://es.wikipedia.org/wiki/HTML_din%C3%A1mico)

2.3.7.4. JavaScript

En el sitio Web de **Mayer Horna García** dice: “Es un lenguaje de programación interpretado (scripting), es decir, que no requiere compilación, se utiliza la creación de páginas web. No es un lenguaje de Programación Orientada a Objetos propiamente dicho como Java, pero gracias a que es basado en prototipos, se puede aplicar conceptos de programación orientada a objetos.” (<http://www.slideshare.net/mayerhorna/01-ext-js-introduccion>)

2.3.7.5. XML (Extensible Markup Languaje)

En el sitio Web de **W3** dice: “Extensible Markup Language (XML) es un formato simple de texto muy flexible derivado de SGML (ISO 8879). Originalmente diseñado para afrontar los retos de las grandes publicaciones electrónicas, XML también está desempeñando un papel cada vez más importante en el intercambio de una amplia variedad de datos en la Web y en otros lugares.”
(<http://www.w3.org/XML/>)

2.3.7.6. CSS (Cascading Style Sheets)

En el sitio Web de **Wikipedia** dice: “Es un lenguaje de hojas de estilo para describir la semántica de la presentación (el aspecto y el formato) de un documento escrito en un lenguaje de marcas . Su aplicación más común es el estilo de las páginas web escritas en HTML y XHTML , pero el lenguaje también se puede aplicar a cualquier tipo de XML documento, incluyendo XML simple , SVG y XUL .DOM (Document Object Model).”
http://en.wikipedia.org/wiki/Cascading_Style_Sheets)

2.3.7.7. DOM (Document Object Model)

En el sitio Web de **Wikipedia** dice: “**DOM** ('Modelo de Objetos del Documento' o 'Modelo en Objetos para la representación de Documentos') es esencialmente una interfaz de programación de aplicaciones (API) que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos. A través del DOM, los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML, que es para lo que se diseñó principalmente.”
(http://es.wikipedia.org/wiki/Document_Object_Model)

2.3.7.8. Ext Js

En el sitio Web de **Wikipedia** dice: “Es una biblioteca de JavaScript para el desarrollo de aplicaciones web interactivas RIA usando tecnologías como AJAX, DHTML CSS y DOM. ExtJS cuenta con licencias open source y comerciales.”

Figura 2.6. Logo de Ext JS



(http://es.wikipedia.org/wiki/Ext_JS)

2.3.8. Navegadores web

En el sitio Web de **Wikipedia** dice: “Un navegador o navegador web (*web browser*) es un programa que permite visualizar la información que contiene una. El navegador interpreta el código, HTML generalmente, en el que está escrita la página web y lo presenta en pantalla permitiendo al usuario interactuar con su contenido y navegar hacia otros lugares de la red mediante enlaces o hipervínculos.” (http://es.wikipedia.org/wiki/Navegador_web)

Entre algunos de los navegadores web tenemos: Safari, Google Chrome, Internet Explorer, Mozilla Firefox, Opera, Amaya, Netscape.

Figura 2.7. Logos de Google Chrome, Internet Explorer y Mozilla Firefox.



Obtenido de: http://es.wikipedia.org/wiki/Navegador_web

2.3.9. Inventarios

En el sitio Web de **Giovanny E. Gómez** dice: “Un inventario representa la existencia de productos terminados que tiene la empresa para comerciar con ellos, comprándolos y vendiéndolos tal cual o procesándolos primero antes de venderlos, en un período económico determinado.

Inventario permanente

Por medio de este sistema la empresa conoce el valor de la mercancía en existencia en cualquier momento, sin necesidad de realizar un conteo físico, por que los movimientos de compra y venta de mercancías se registra directamente en el momento de realizar la transacción a su precio de costo.

Las empresas que adoptan este tipo de sistema deben llevar un auxiliar de mercancías denominado "Kárdex", en el cual se registra cada artículo que se compre o que se venda. La suma y la resta de todas las operaciones en un periodo da como resultado el saldo final de mercancías.

Métodos para la valoración de inventarios

Los métodos más utilizados para fijar el costo de las mercancías de la empresa son el promedio ponderado, UEPS o FIFO y PEPS o LIFO, a continuación se presentan sus fundamentos y un ejemplo de su aplicación:

- Método del promedio ponderado: Este método consiste en hallar el costo promedio de cada uno de los artículos que hay en el inventario final cuando las unidades son idénticas en apariencia, pero no en el precio de adquisición, por cuanto se han comprado en distintas épocas y a diferentes precios.

Para fijar el valor del costo de la mercancía por este método se toma el valor de la mercancía del inventario inicial y se le suman las compras del periodo, después se divide por la cantidad de unidades del inventario inicial más las compradas en el periodo.

- Método PEPS o FIFO: Aplicándolo a las mercancías significa que las existencias que primero entran al inventario son las primeras en salir del mismo, esto quiere decir que las primeras que se compran, son las primeras que se venden.
- Método UEPS o LIFO: Este método tiene como base que la última existencia en entrar es la primera en salir. Esto es que los últimos adquiridos son los primeros que se venden.”

(<http://www.gestiopolis.com/canales/financiera/articulos/14/inventarios.htm>)

Figura 2.8. Kardex de inventario

EMPRESA XYZ										
MERCANCIAS: Radio			REFERENCIA: E-10980			METODO: Promedio Ponderado				
ARTICULO: Radio marca Fénix						CÓDIGO: 12237737				
FECHA	DETALLE	ENTRADAS			SAUDA			EXISTENCIAS		
		C	V.U	V.T	C	V.U	V.T	C	V.U	V.T
01/10/2010	Inventario inicial							100	100	10000,00
02/10/2010	Compra Factura N 450	30	120	3600,00				130	104,62	13600,60
03/10/2010	Venta Factura N 120				40	104,62	4184,80	90	104,61	9414,90
04/10/2010	Venta Factura N 121				20	104,61	2092,20	70	104,61	7322,70
05/10/2010	Devolución Venta Factura N 121				-10	-104,61	-1046,10	80	104,61	8368,80
06/10/2010	Compra Factura N 850	60	140	8400,00				140	119,78	16769,20
07/10/2010	Devolución compra Factura N 850	-15	-140	-2100,00				125	117,35	14668,75
08/10/2010	Venta Factura N 122				60	117,35	7041,00	65	117,36	7628,40
SALDO FINAL								65		7628,40

2.3.10. Facturación

En el sitio Web de **Wikipedia** dice: “Una factura es un documento que respalda la realización de una operación económica, que por lo general, se trata de una compraventa.

La información fundamental que aparece en una factura debe reflejar la entrega de un producto o la provisión de un servicio, junto a la fecha de devengo, además de indicar la cantidad a pagar como contraprestación.”

(<http://es.wikipedia.org/wiki/Factura>)

2.3.11. Administración

Según, Administración, teórica, proceso y práctica. Colombia, McGraw - Hill Interamericaba, S.A.; Pag.: “La administración es el proceso de planificar, organizar, dirigir y controlar el uso de los recursos organizacionales para conseguir determinados objetivos con eficiencia y eficacia. “

2.3.12. Contabilidad

Según, Temas de contabilidad básica e intermedia. 4ta Edición; Editorial Educación y Cultura; Pág 4.: “La contabilidad es un instrumento de comunicación de hechos económicos, financieros y sociales suscitados en una empresa, sujeto a medición, registración e interpretación para la toma de decisiones empresariales.

2.4. HIPÓTESIS

El desarrollo del módulo de gestión de inventario y facturación bajo arquitectura SOA permitirá optimizar el Sistema Administrativo Contable de la empresa BESIXPLUS CIA. LTDA.

2.5. SEÑALAMIENTO DE LAS VARIABLE DE LA HIPÓTESIS

2.5.1. Variable independiente

Módulo de gestión de inventario y facturación bajo arquitectura SOA.

2.5.2. Variable dependiente

Sistema administrativo contable.

CAPÍTULO III

METODOLOGÍA

3.1. ENFOQUE

La presente investigación tiene un enfoque cuali – cuantitativo porque participa personal de la empresa para solucionar sus necesidades y problemas bajo la guía de un técnico, para posteriormente ser el investigador quien compruebe la hipótesis basándose en la información obtenida.

3.2. MODALIDAD DE LA INVESTIGACIÓN

3.2.1. Investigación de Campo

Esta investigación nos permite recolectar información en el mismo lugar de los hechos, visitaremos la empresa BESIXPLUS CIA LTDA., para observar, inspeccionar y entrevistar a los empleados, de esta manera obtener información que nos permita adquirir elementos de juicio y poder evaluar la situación real de la empresa pudiendo proponer soluciones que ayuden a cambiar la situación actual de la misma

3.2.2. Investigación Bibliográfica – Documental

Esta modalidad nos permite conocer, desarrollar, ampliar, profundizar, y deducir diferentes enfoques, teorías, conceptualizaciones y criterios de diversos autores sobre una cuestión determinada basándose en internet(fuentes primarias) o en libros, revistas, periódicos y otras publicaciones(fuentes secundarias).Lo cual nos

permitirá poder respaldar los resultados que se obtendrán de la investigación en la empresa BESIXPLUS CIA LTDA.

3.2.3. Proyecto Factible

Se desarrollará una propuesta en base a la construcción del módulo de gestión de inventario y facturación para el Sistema Administrativo Contable, el cual es un modelo práctico que permitirá solucionar los problemas detectados en la empresa BESIXPLUS CIA LTDA. previo al diagnóstico realizado con anterioridad y sustentado en el marco teórico.

3.3. NIVEL O TIPO DE INVESTIGACIÓN

Para la realización de la presente investigación se manejará un nivel de investigación exploratorio para detectar las características del problema y determinar si es factible o no de solucionarse. Pasaremos a un nivel descriptivo el cual permitirá representar de una manera clara tal y cómo se manifiesta el fenómeno en estudio. Por último llegar a un nivel operacional que nos facilitará la comprensión y análisis del fenómeno dentro de un contexto determinado para fundamentar las bases teóricas.

3.4. POBLACIÓN Y MUESTRA

3.4.1. Población

En la presente investigación la población objeto de estudio, está constituida por 6 personas.

3.4.2. Muestra

Como la población es muy pequeña pasará a constituir la muestra.

3.5. OPERACIONALIZACIÓN DE LAS VARIABLES

3.5.1. Variable independiente

Tabla 3.1. Operacionalización de variables – Variable independiente

Conceptualización	Dimensión	Indicadores	Ítems	Técnica
Módulo de gestión de inventario y facturación bajo arquitectura SOA: Conjunto de procedimientos, métodos y técnicas para organizar y controlar el flujo de información de una empresa. Utilizando tecnologías web.	Inventario	Administración actual de inventarios	¿Qué sistema de inventario utiliza para el control de movimientos de la mercadería?	Encuesta Inspección
	Facturación	Conocimientos de facturación.	¿Cómo califica el procedimiento utilizado en la Empresa para el control de mercadería?	Encuesta Observación
			¿Conoce usted el procedimiento para elaborar una factura?	Encuesta Observación
	Tecnologías web	Web	¿Conoce usted las disposiciones reglamentarias emitidas por el SRI relacionadas a los comprobantes de venta?	Encuestas
			¿Cuáles piensa usted que serían las ventajas de que el sistema de inventarios y facturación se utilice a través de internet?	Entrevista

3.5.2. Variable dependiente

Tabla 3.2. Operacionalización de variables – Variable dependiente

Conceptualización	Dimensión	Indicadores	Ítems	Tec-Inst
<p>Sistema administrativo contable: El sistema de información contable es la combinación del personal, los registros y los procedimientos que se usan en un negocio para cumplir con las necesidades de información financiera.</p>	Contabilidad	Operaciones comerciales	¿Cómo califica usted el servicio al cliente?	Encuesta Observación
		Documentos comerciales	¿La información de inventarios se encuentra almacenada en algún tipo de base de datos?	Encuesta Inspección
	Administración	Eficiencia del personal	¿Cuál de estos paquetes contables ha manejado usted?	Encuesta
		Empresa	¿Existe en su Empresa elaborado un manual de funciones?	Encuesta
		Empresa	¿Cómo califica el crecimiento en ventas de su Empresa?	Encuesta

3.6. RECOLECCIÓN DE INFORMACIÓN

3.6.1. Plan de recolección de información

Se reunirá información de fuentes bibliográficas, se aplicaran entrevistas estructuradas al personal encardado de bodega y ventas y al personal directivo de BESIXPLUS CIA LTDA.

3.6.2. Plan de procesamiento y análisis de la información

Para analizar y procesar la información de la presente investigación se reunirá, presentará y resumirá todos los datos que han sido recogidos mediante la técnica señalada para el efecto. Luego se procederá a la clasificación de los datos recopilados, se tabularán esto en forma manual por ser un método simple, rápido y efectivo, siempre y cuando se mantenga el debido cuidado para no alterar la información.

CAPÍTULO IV

ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

4.1. ANÁLISIS E INTERPRETACIÓN DE LOS RESULTADOS

Para el análisis e interpretación de resultados, cabe destacar que las entrevistas (ANEXO 1), se realizaron a los 6 miembros de la junta directiva de la empresa BESIXPLUS CIA. LTDA.

4.1.1. Análisis de los resultados de las entrevistas

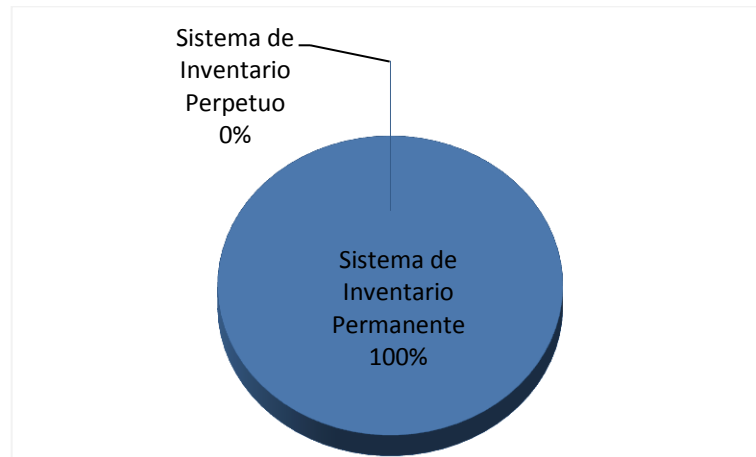
Pregunta 1

¿Qué Sistema de Inventario utiliza para el control de movimientos de la mercadería?

Tabla 4.1. Cuadro de Frecuencia Pregunta 1

X Alternativa	F Frecuencia	F^c %
Sistema de Inventario Permanente	6	100%
Sistema de Inventario Perpetuo	0	0%
Total	6	100%

Figura 4.1. Sistema de Inventario utilizado.



Análisis e interpretación

El 100% de los encuestados afirman que el sistema de inventarios utilizado para el control de mercadería en la Empresa es el Sistema de Inventarios Permanente

En la inspección realizada por la investigadora se pudo constatar que el control de movimientos de mercaderías se hace a través del Sistema de inventarios permanente el cual permite llevar un registro de cada unidad que ingresa o sale del inventario, además se pudo comprobar que éste proceso se lo realiza de forma manual.

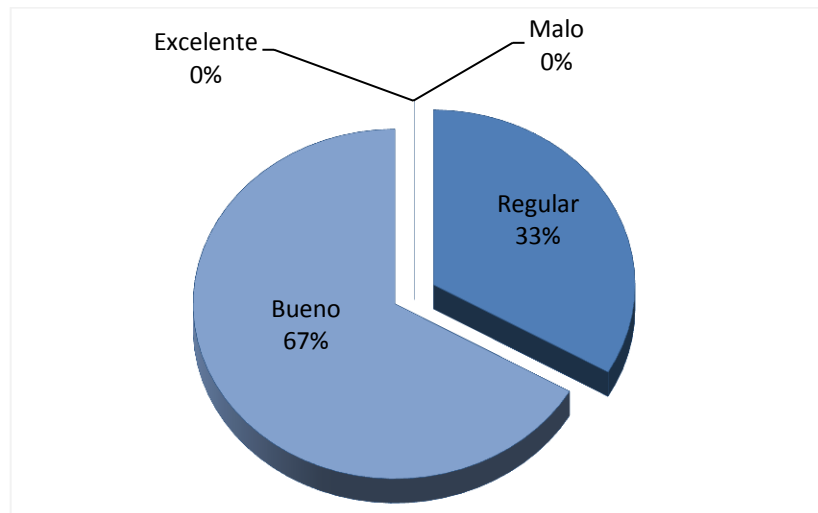
Pregunta 2

¿Cómo califica el procedimiento utilizado en la Empresa para el control de mercadería?

Tabla 4.2. Cuadro de Frecuencia Pregunta 2

X Alternativa	F Frecuencia	F' %
Malo	0	0%
Regular	2	33%
Bueno	4	67%
Excelente	0	0%
Total	6	100%

Figura 4.2. Forma de control de mercadería.



Análisis e interpretación

El 67% de los encuestados opina que el procedimiento utilizado en la Empresa para el control de mercadería es bueno; el 33% dice que es regular.

En investigación de campo la investigadora pudo notar que los empleados de la Empresa BESIXPLUS CIA. LTDA. realizan un buen trabajo en el control de mercadería pero indican que es necesario implantar un sistema para que se maneje automáticamente la información para mantenerla actualizada y sin errores

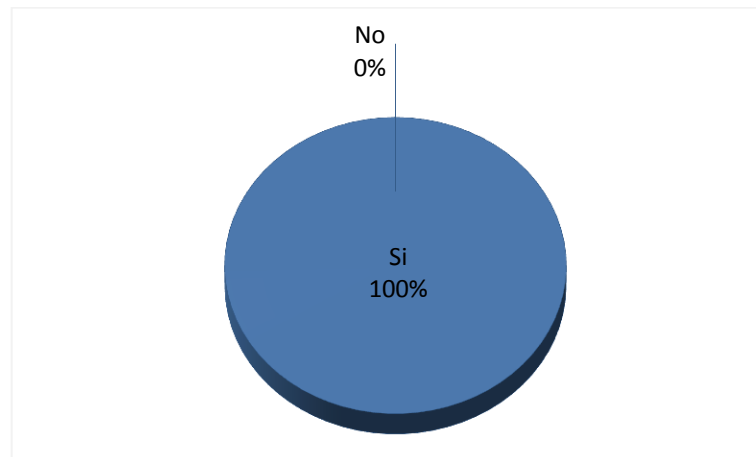
Pregunta 3

¿Conoce usted el procedimiento para elaborar una factura?

Tabla 4.3. Cuadro de Frecuencia Pregunta 3

X Alternativa	F Frecuencia	F' %
Si	6	100%
No	0	0%
Total	6	100%

Figura 4.3. Procedimiento para elaborar facturas



Análisis e interpretación

Sobre el conocimiento del personal de la Empresa BESIXPLUS CIA. LTDA. acerca del procedimiento para elaborar una factura el 100% opina que se encuentran capacitado para realizar esta tarea.

Se pudo observar que evidentemente el personal tiene conocimiento y está capacitado para elaborar una factura.

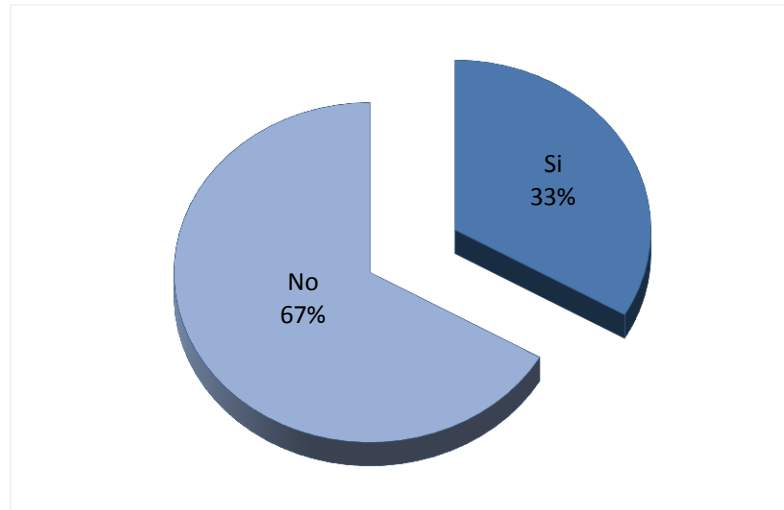
Pregunta 4

¿Conoce usted las disposiciones reglamentarias emitidas por el SRI relacionadas a los comprobantes de venta?

Tabla 4.4. Cuadro de Frecuencia Pregunta 4

X Alternativa	F Frecuencia	F' %
Si	2	33%
No	4	67%
Total	6	100%

Figura 4.4. Conocimiento de reglamentos del SRI.



Análisis e interpretación

De acuerdo al conocimiento de los reglamentos del SRI referente a los comprobantes de venta el 67% de las personas encuestadas no tienen conocimiento de dichos reglamentos mientras el 33% restante indica que tiene conocimiento pero no de forma completa y específica.

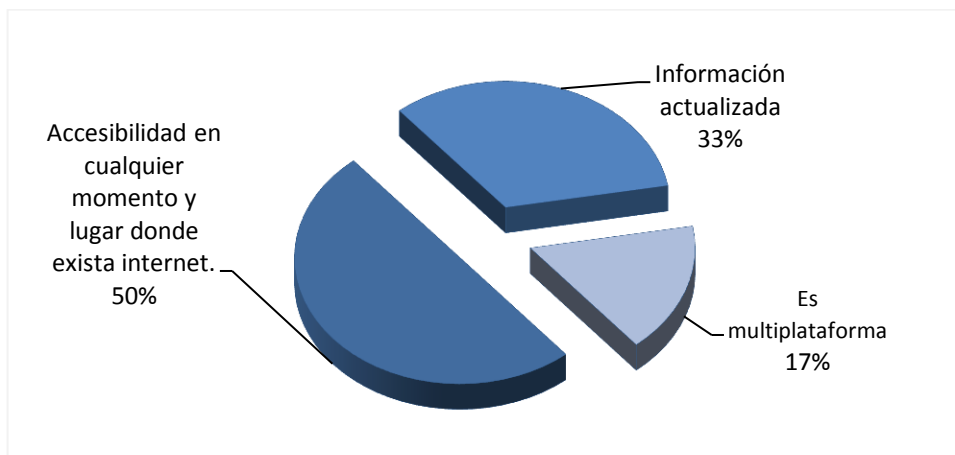
Pregunta 5

¿Cuáles piensa usted que serían las ventajas de utilizar el sistema de inventarios y facturación a través de internet?

Tabla 4.5. Cuadro de Frecuencia Pregunta 5

X Respuestas	F Frecuencia	F' %
Accesibilidad en cualquier momento y lugar donde exista internet.	3	50%
Información actualizada	2	33%
Es multiplataforma	1	17%
Total	6	100

Figura 4.5. Ventajas de utilizar el sistema a través de internet



Análisis e interpretación

Las personas entrevistadas opinan una de las principales ventajas de utilizar el sistema de inventario y facturación a través de internet es la accesibilidad obteniendo el 50% entre las principales ventajas; el 33% de las personas indican que se debería utilizar por internet ya que permitirá tener la información actualizada y disponible para todo el personal de la Empresa; y el 17% opina que se puede utilizar en cualquier plataforma operativa.

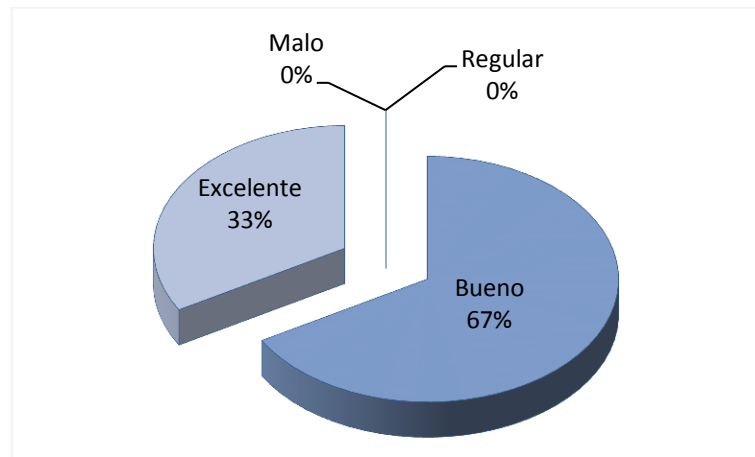
Pregunta 6

¿Cómo califica usted el servicio al cliente?

Tabla 4.6. Cuadro de Frecuencia Pregunta 6

X Alternativa	F Frecuencia	F' %
Malo	0	0%
Regular	0	0%
Bueno	4	67%
Excelente	2	33%
Total	6	100%

Figura 4.6. Servicio al cliente



Análisis e interpretación

Sobre el servicio al cliente el personal de la Empresa BESIXPLUS CIA. LTDA. el 67% opina que el servicio al cliente es bueno debido a que la información referente a mercadería no se encuentra a la mano y/o actualizada lo cual evita que se preste un servicio ágil; el 33% restante opina que la atención es excelente.

En investigación de campo la investigadora pudo notar el servicio al cliente es bueno pero que necesita mejorar.

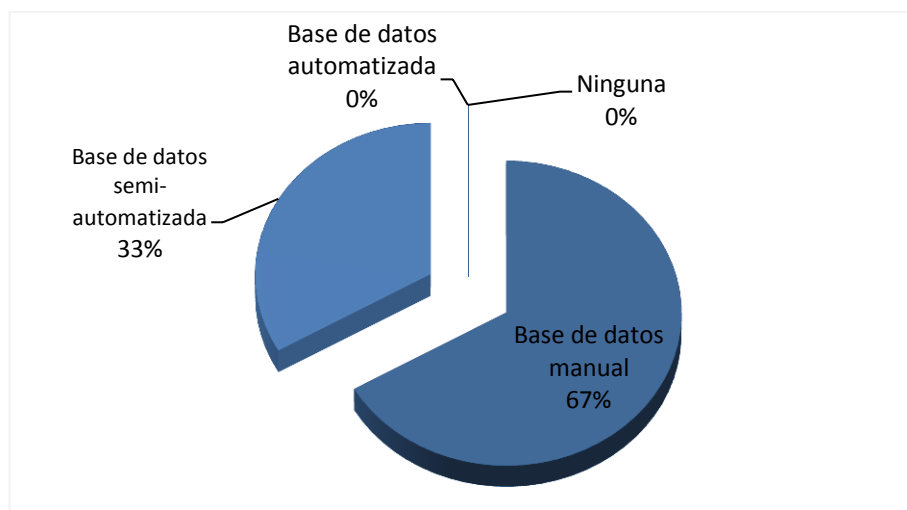
Pregunta 7

¿La información de inventarios se encuentra almacenada en algún tipo de base de datos?

Tabla 4.7. Cuadro de Frecuencia Pregunta 7

X Respuestas	F Frecuencia	F' %
Base de datos manual	4	67%
Base de datos semi-automatizada	2	33%
Base de datos automatizada	0	0%
Ninguna	0	0%
Total	6	100%

Figura 4.7. Almacenamiento de la información



Análisis e interpretación

De acuerdo al almacenamiento de la información de los inventarios el 67% de las personas encuestadas opinan que esta información se almacena en una base de datos manual mientras el 33% restante indica que el almacenamiento es en una base de datos semi-automatizada.

De la inspección realizada por la investigadora en la Empresa BESIXPLUS CIA. LTDA. se pudo concluir que la información de inventarios se almacenan en un archivero y también en hojas de cálculo de Microsoft Excel.

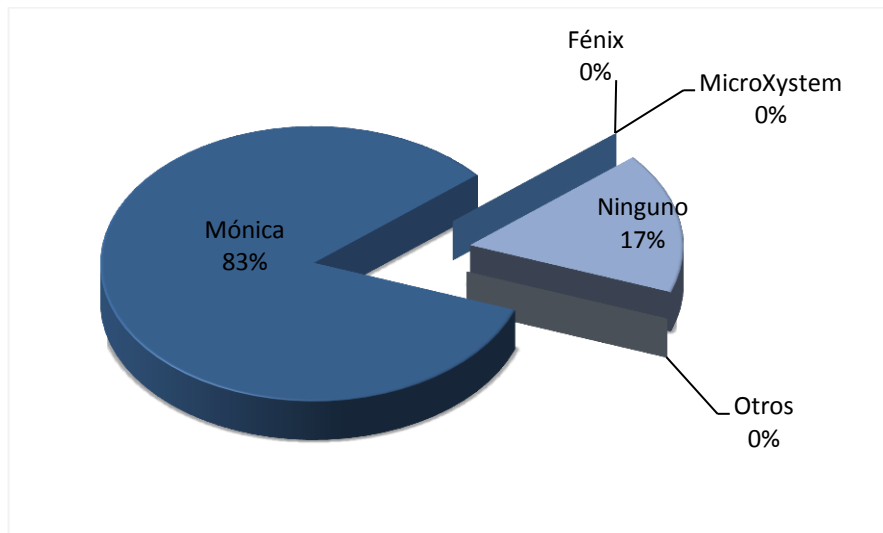
Pregunta 8

¿Cuál de estos paquetes contables ha manejado usted?

Tabla 4.8. Cuadro de Frecuencia Pregunta 8

X Respuestas	F Frecuencia	F' %
Mónica	5	83%
Fénix	0	0%
MicroXsystem	0	0%
Ninguno	1	17%
Otros	0	0%
Total	6	100%

Figura 4.8. Manejo de paquetes contables



Análisis e interpretación

En la Empresa BESIXPLUS CIA. LTDA. el 83% del personal indica que ha utilizado el paquete contable Mónica pero de forma parcial y hace mucho tiempo atrás; mientras el 17% no ha utilizado ninguno paquete contable.

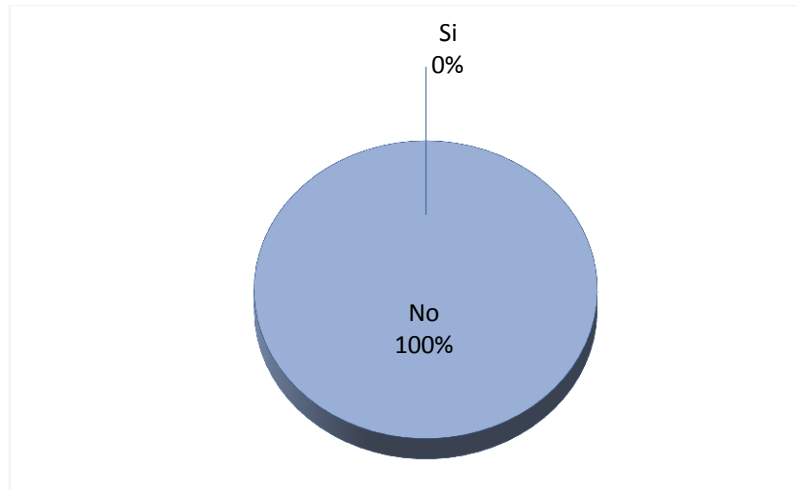
Pregunta 9

¿Existe en su Empresa elaborado un manual de funciones?

Tabla 4.9. Cuadro de Frecuencia Pregunta 9

X Alternativa	F Frecuencia	F' %
Si	0	0%
No	6	100%
Total	6	100%

Figura. 4.9. Manual de funciones.



Análisis e interpretación

En la Empresa BESIXPLUS CIA. LTDA. el 100% del personal indica que no existe un manual de funciones que permita delimitar las actividades de cada empleado.

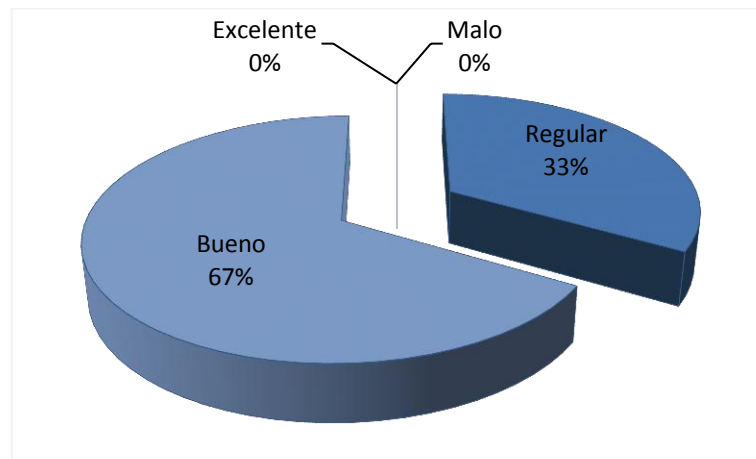
Pregunta 10

¿Cómo califica el crecimiento en ventas de su Empresa?

Tabla 4.10. Cuadro de Frecuencia Pregunta 10

X Alternativa	F Frecuencia	F' %
Malo	0	0%
Regular	2	33%
Bueno	4	67%
Excelente	0	0%
Total	6	100%

Figura 4.10. Crecimiento de ventas en la Empresa



Análisis e interpretación

Acerca del crecimiento en ventas en la Empresa BESIXPLUS CIA. LTDA. el 67% del personal indica que ha sido bueno y el 33% restante opina que ha sido regular.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- El control de las mercaderías se lo realiza mediante el Sistema de Inventarios Permanente utilizando tarjetas kardex, el registro de movimientos de inventario se lo hace manualmente debido a esto la información puede ser alterada en cualquier momento, estos factores son sin duda un elemento crítico para el buen desarrollo de BESIXPLUS CIA. LTDA.
- El personal conoce el procedimiento para elaborar una factura pero desconoce sobre los reglamentos emitidos por el SRI referente a los comprobantes de venta, lo cual ha provocado que el proceso administrativo se lleve de forma incorrecta, además la elaboración de las facturas se lo hace de forma manual.
- En la Empresa se ha podido observar que la información de los inventarios se encuentran almacenados en archiveros y hojas de cálculo, en algunos casos los datos son inservibles por razones de duplicidad o antigüedad evitando así obtener información de forma ágil y precisa.
- BESIXPLUS CIA. LTDA. no cuenta con un manual de funciones que permita delimitar las tareas del personal esto ha evitado mejorar la calidad de sus servicios y cumplir de mejor manera con los objetivos propuestos

5.2 RECOMENDACIONES

- Desarrollar un sistema informático que maneje automáticamente los procesos de inventario y facturación para facilitar el trabajo diario de las personas relacionadas con estas actividades.
- Debido a que no todo el personal de la Empresa cuenta con conocimientos sobre los reglamentos del SRI referente a los comprobantes de venta, deberá organizarse y planearse la mejor manera en que estos pueden capacitarse.
- La aplicación a desarrollarse deberá ser accesible desde cualquier parte del mundo sin importar la plataforma operativa que se esté utilizando, además debe ser de fácil uso, rápido y seguro ya que la información que se maneja es de gran importancia para la Empresa.
- Realizar una amplia investigación sobre las tecnologías informáticas con las que cuenta la Empresa para la implementación e implantación del Sistema, además se recomienda utilizar herramientas de software libre para el desarrollo de la aplicación ya que son herramientas confiables, potentes y eficientes.
- Realizar informes detallados con los objetivos, alcance y requerimientos del sistema a desarrollarse para así evitar problemas con el resultado final y evitar sobre tiempos en la entrega del proyecto.
- Es importante que la Empresa BESIXPLUS CIA. LTDA elabore un manual interno de funciones.

CAPÍTULO VI

PROPUESTA

6.1. DATOS INFORMATIVOS

Institución Ejecutoriada: Empresa BESIXPLUS CIA. LTDA

Beneficiarios: Empresa BESIXPLUS CIA. LTDA

Tiempo: El presente proyecto va a ser ejecutado en el período noviembre 2010 - noviembre 2011.

Costo: El costo estimado para desarrollar el proyecto es de \$2417.50 dólares americanos. (ANEXO 2)

6.2. ANTECEDENTES DE LA PROPUESTA

La empresa BESIXPLUS CIA. LTDA. de la ciudad de Ambato constituida legalmente el 28 de Abril del 2006, pero ha venido desarrollando sus funciones desde hace tiempo atrás y cabe indicar que en todo este período no se ha realizado un estudio sobre la necesidad de implantar un sistema informático para el manejo de inventario y facturación.

En la Empresa se ha podido observar que los procesos de inventario y facturación se realizan manualmente, la información se encuentra almacenada en archiveros y

hojas de cálculo no se cuenta con información actualizada lo cual evita que se preste un servicio ágil y de calidad a los clientes.

Debido a esta necesidad se ha optado por desarrollar un sistema informático para la gestión de inventario y facturación el mismo que será desarrollado con herramientas de software libre, utilizando la arquitectura de software SOA basado en servicios web, lenguaje de programación java, SGBD PostgreSql y para el desarrollo de la interfaz de usuario se utilizara extjs, javaScript y ajax el sistema será orientado a la web proponiendo múltiples ventajas económicas y tecnológicas para la empresa.

6.3. JUSTIFICACIÓN

El desarrollo de la presente propuesta de investigación se fundamenta en la necesidad de atender de mejor manera, amplia y oportuna las necesidades del cliente y de la Empresa a través de la automatización los procesos de inventario y facturación.

El manejar los procesos de inventario y facturación de manera automatizada permitirá mantener la información ordenada y al día para poder utilizarla en la toma de decisiones, conocer con exactitud cantidades y productos existentes en bodega y saber con precisión que productos y/o servicios son requeridos con mayor frecuencia por los clientes.

Igualmente, la ejecución de la presente propuesta es factible y viable para ser utilizada en la empresa BESIXPLUS CIA. LTDA, como una herramienta administrativa para el manejo del inventario y facturación.

6.4. OBJETIVOS

6.4.1. Objetivo General

Desarrollar un sistema informático para la gestión de inventario y facturación utilizando herramientas de software libre aplicando la arquitectura de software SOA basada en servicios web.

6.4.2. Objetivos Específico

- Analizar y definir los requerimientos del módulo de gestión de inventario y facturación.
- Diseñar e implementar el modelo de base de datos en base a los requerimientos recopilados.
- Desarrollar el módulo de gestión de inventario y facturación con una interfaz amigable y de fácil manejo para los usuarios.
- Realizar pruebas del sistema para garantizar y verificar la validez de la información.
- Capacitar al administrador del sistema sobre el funcionamiento e instalación de la aplicación.

6.5. ANÁLISIS DE FACTIBILIDAD

6.5.1. Factibilidad operativa

El sistema fue desarrollado con una interfaz amigable y de fácil manipulación para el usuario, pudiendo realizar varias tareas al mismo tiempo ya que la interfaz es similar al entorno de Windows XP. Además el sistema a desarrollarse será mantenible ya que es susceptible a cambios, confiable por que se maneja un alto nivel de seguridad y se controla fallos tanto en base de datos como en el cliente, y eficiente ya que se utiliza de una mejor manera los recursos de la empresa.

El usuario podrá realizar todas las tareas referentes a facturación e inventario, dependiendo del perfil de usuario que tenga asignado.

El sistema permitirá llevar un control correcto y organizado de las entradas y salidas de mercaderías atreves de la kardex de inventario, esta herramienta permitirá imprimir reportes con información resumida de las transacciones de inventario registradas. Puede realizar un seguimiento de movimientos y costos de los productos.

Mediante la opción de facturación se podrá facturar productos tangibles e intangibles, aplicar descuentos por productos y/o al total de la venta. Realizar ventas a contado o crédito. Además puede generar reportes sobre las ventas realizadas en general o por filtros.

6.5.2. Factibilidad económica

Al desarrollar el proyecto utilizando herramientas de software libre tanto en lenguaje de programación como en el gestor de base de datos, el ahorro económico es muy significativo ya que la empresa evita invertir en la compra de licencias.

Se optimizará la gestión del inventario y facturación lo cual lleva al mejoramiento económico de la empresa.

6.5.3. Factibilidad técnica

Para el desarrollo del módulo de gestión de inventario y facturación se ha optado por utilizar herramientas de software libre.

Software

- Lenguaje de programación: Java
- Motor de base de datos: PostgreSQL 9.0
- Interfaz de usuario: Extjs 3.3.1, javaScript, ajax, html
- Navegador web: Mozilla firefox 3.6 o superior , Google Chrome
- Servidor de aplicaciones: JBoss 5.1.0 GA
- Servidor de reportes: JasperServer 3.7.0

Hardware para el cliente

- CPU 900Mhz o Superior.
- RAM 1024Mb o Superior.
- Conexión a red informática.

6.6. FUNDAMENTACIÓN

EtxJs

ExtJS es una librería Javascript que permite construir aplicaciones complejas en internet además de flexibilizar el manejo de componentes de la página como el DOM, peticiones AJAX, DHTML, tiene la gran funcionalidad de crear interfaces de usuario bastante funcionales.

Esta librería incluye:

- Componentes UI del alto performance y personalizables.
- Modelo de componentes extensibles.
- Un API fácil de usar.
- Licencias Open Source (GPL) y comerciales

JSON

Acrónimo de *JavaScript Object Notation*, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML.

La simplicidad de JSON ha dado lugar a la generalización de su uso, especialmente como alternativa a XML en AJAX. Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador sintáctico (parser) de JSON. En JavaScript, un texto JSON se puede analizar fácilmente usando el procedimiento `eval()`, lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la comunidad de desarrolladores AJAX, debido a la ubicuidad de JavaScript en casi cualquier navegador web.

Firebug

Es una extensión de Firefox creada y diseñada especialmente para desarrolladores y programadores web. Es un paquete de utilidades con el que se puede analizar

(revisar velocidad de carga, estructura DOM), editar, monitorizar y depurar el código fuente, CSS, HTML y JavaScript de una página web de manera instantánea e inline.

Firebug no es un simple inspector como DOM Inspector, además edita y permite guardar los cambios, un paso por delante del conocido Web Developer. Su atractiva e intuitiva interfaz, con solapas específicas para el análisis de cada tipo de elemento (consola, HTML, CSS, Script, DOM y red), permite al usuario un manejo fácil y rápido. Firebug está encapsulado en forma de plug-in o complemento de Mozilla, es Open Source, libre y de distribución gratuita.

6.7. METODOLOGÍA

Figura 6.1. Metodología XP



Una metodología es un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores transformar los requisitos del usuario en un sistema software.

La metodología de desarrollo de software que se utiliza para este sistema es Extreme Programming (XP) que consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

Bases de la metodología XP

Pruebas Unitarias: se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, podamos hacer pruebas de las fallas que pudieran ocurrir.

- Refabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- Programación en pares: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.

Para estructurar el proceso de diseño de tal forma que los analistas, clientes, desarrolladores y otras personas involucradas en el desarrollo del sistema lo comprendan se utilizó como herramienta de diseño al lenguaje de modelado UML que es un lenguaje gráfico para el modelado orientado a objetos con el que se puede pensar, visualizar, analizar, comunicar y registrar. Facilita la comunicación a través de fases, versiones y generaciones.

Además es un estándar abierto el cual soporta todo el ciclo de vida de desarrollo de software como es el análisis y la especificación de requerimientos, diseño, implementación, implantación y pruebas, soporta todo tipo de arquitectura de software y lenguaje de programación.

6.8. MODELO OPERATIVO

6.8.1. Análisis del sistema

6.8.1.1. Análisis y requerimientos del sistema

De la investigación realizada en BESIXPLUS CIA. LTDA. y bajo la coordinación del jefe de proyecto en la empresa se ha establecido los requerimientos con los cuales cumplirá el módulo de gestión de inventarios y facturación.

Uno de los requerimientos de mayor importancia es que el kardex de inventario se pueda manejar con los tres diferentes métodos: promedio ponderado, fifo y lifo. Se pueda realizar ventas a contado y crédito, verificando que el cliente tenga una solicitud de crédito aprobada y validando el crédito máximo que tiene con la Empresa.

Se definirán campos de auditoría en cada una de las tablas en los que se registrará información del usuario y la fecha en la cual se realiza la inserción o actualización de los datos.

A continuación se detalla el listado de requerimientos:

Requerimientos funcionales

Tarjetas kardex

- Se podrá registrar movimientos de entrada y salida de productos directamente en el kardex o automáticamente a través de los procesos que genera la facturación, compras y devolución de ventas y compras.
- Se podrá utilizar el método promedio ponderado.
- Se podrá registrar las series de los productos.
- Se registrará los valores de los productos a precio de costo.

Restricciones

- No se podrá modificar los datos del kardex una vez guardada la información
- No se podrá eliminar movimientos registrados

Observaciones

- El método a utilizar en el kardex dependerá de las políticas de la empresa.

Toma y ajuste de inventario

- Se realizarán cortes de inventarios para efectuar tomas físicas por muestreo en cualquier momento.

- Se podrá localizar el producto en la bodega así como el stock disponible que hay del mismo.
- Ingreso de cantidades encontradas en el conteo físico.
- Informes de diferencias en inventarios obtenidos en el conteo físico con los inventarios del sistema.
- Ajustes de diferencias de inventarios automatizados de tal forma que afecte tanto al kárdex del producto como al registro contable

Restricciones

- No se podrá realizar un cierre de inventario cuando otro este abierto.
- No se podrá hacer cierres de inventario mayores a la fecha actual.
- Los ajustes de inventario se harán de cierres generados en estado abierto.

Observaciones

- Las fechas para cierres de inventario dependerán de las políticas de la empresa.

Facturación

- La facturación será realizada de forma interactiva. Se le preguntará al cliente sus datos personales y los productos y/o servicios que desea adquirir (pueden ser varios).
- Para la facturación se consultarán los datos del cliente el cual debe estar previamente registrado en el módulo de clientes.
- Los productos que forman parte del detalle de la factura deberán estar previamente registrados en el módulo de compras e inventario.
- Se podrá aplicar o no descuentos al total de la factura y a los productos y/o servicios de forma individual.

- La aplicación permitirá aplicar dos formas de pago Crédito y Contado.
- Se deberá comprobar que si el pago es a Crédito la cantidad no debe sobrepasar el límite máximo de crédito asignado al cliente.
- Los movimientos que genera la venta se registrarán en el módulo de contabilidad e inventario.
- Se podrá generar una copia impresa de la factura con el formato del documento de la empresa.

Restricciones

- No se podrá modificar la factura una vez que haya sido guardada.
- No se podrá modificar el porcentaje de IVA directamente en la factura.
- No se podrá modificar los precios de los productos o servicios directamente en el detalle de la factura.
- No se podrá modificar el nombre del vendedor.

Observaciones

- La persona a solicitar el crédito deberá estar registrada en el módulo de clientes y tener una solicitud de crédito aprobada. El tiempo, formas de pago y montos máximos de créditos serán definidas según políticas de la empresa.
- El porcentaje de IVA deberá ser previamente configurado como un parámetro general.

Devolución de ventas (notas de crédito)

- La devolución será realizada de forma interactiva. Se solicitará al cliente el comprobante de venta y los productos a devolver (pueden ser varios).

- Para realizar una devolución se verificará que la factura este dentro de la fecha máxima de devolución
- Se deberá controlar las cantidades y series de los productos a devolver sean menores e iguales a las cantidades facturadas.
- Los movimientos que genera la devolución se registrarán en el módulo de contabilidad e inventario.
- Se podrá generar una copia impresa del comprobante con el formato del documento de la empresa.
- Se podrá anular o disminuir el crédito de la venta si esta fue facturada a crédito.

Restricciones

- Una vez guardada la información solo se podrá modificar los datos adjuntos de la devolución.
- Las notas de crédito se emitirán a nombre de la persona que se encuentra registrada la factura y no será modificable.
- Solo se puede devolver productos que se encuentren registrados en el detalle de la factura y no se podrá alterar su precio de venta.
- Los datos de IVA y descuentos que se encuentren registrados en la factura pasaran a ser parte de la nota de crédito y no se podrá modificar.

Observaciones

- El tiempo máximo para la devolución deberá ser previamente configurado como un parámetro general.

Retenciones

- El registro de los datos de la retención será realizado por el usuario del sistema

- Los tipos y porcentajes de retención dependerán del tipo de contribuyente.
- Los movimientos que genera el ingreso de la retención se registrarán en el módulo de contabilidad.
- El valor de la retención deberá afectar al total de la venta.

Observaciones

- El tipo y porcentaje de impuestos dependerán del SRI.
- El tiempo máximo para el registro de retenciones dependerá de políticas de la empresa.
- El vendedor podrá emitir el comprobante de venta menos el valor de la retención siempre y cuando constate la entrega del comprobante por parte del cliente.

Proformas

- La proforma será realizada de forma interactiva. Se le preguntará al cliente sus datos personales y los productos y/o servicios que desea adquirir (pueden ser varios).
- Se podrá emitir proformas a clientes ya registrados o personas particulares.
- Los productos que forman parte del detalle de la proforma deberán estar previamente registrados en el módulo de compras e inventario.
- Se podrá detallar cantidades superiores a las existentes en bodega.
- Se podrá generar una copia impresa de la proforma con el formato del documento de la empresa.

Restricciones

- No se podrá modificar el nombre del cliente una vez que la proforma haya sido guardada.
- No se podrá modificar el porcentaje de IVA directamente en la proforma.

- No se podrá modificar los precios de los productos o servicios directamente en el detalle de la proforma.
- No se podrá modificar el nombre del vendedor.

Requerimientos no funcionales

- Será 100% web y deberá funcionar en distintos navegadores y plataformas operativas.
- Se deberá utilizar como lenguaje de programación java y servicios web.
- La interfaz de usuario se deberá desarrollarse utilizando librerías extjs, javaScript y ajax
- Deberá funcionar sobre PostgreSQL.
- Se utilizará como servidor de aplicaciones jBoss y como servidor de reportes JasperServer

6.8.1.2. Diagramas UML

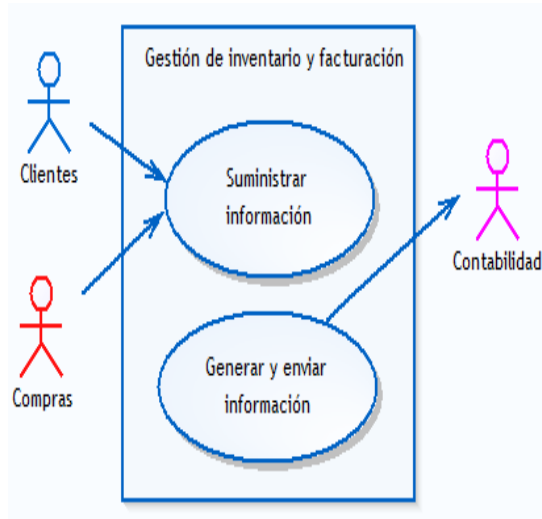
Diagramas de casos de uso

Un diagrama de casos de uso muestra, los distintos requisitos funcionales que se esperan de una aplicación o sistema y cómo se relaciona con su entorno (usuarios u otras aplicaciones).

- **Caso de uso: Interacción del Módulo de gestión de inventario y facturación y los módulos de clientes, compras y contabilidad.**

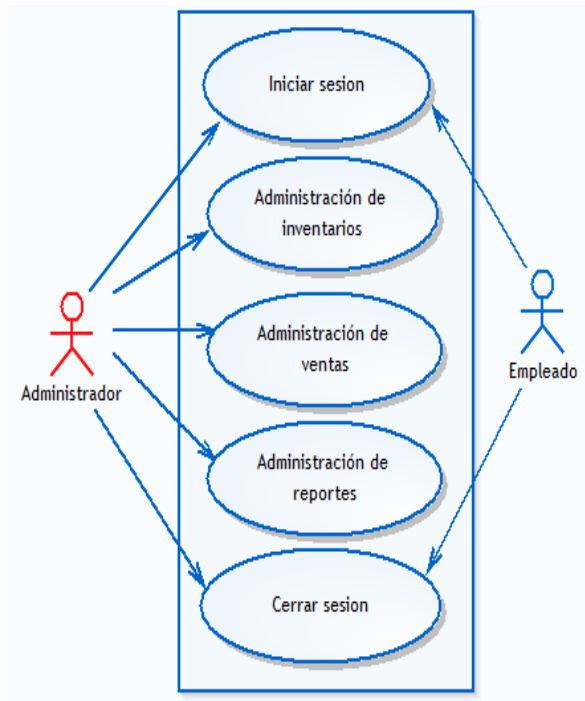
Este caso de uso general Figura 6.2 representa la interacción del módulo de gestión de inventario y facturación y los módulos de clientes, compras y contabilidad.

Figura 6.2. Diagrama de caso de uso de Interacción del Módulo de gestión de inventario y facturación y los módulos de clientes, compras y contabilidad.



- **Caso de uso: Gestión de inventario y facturación**

Figura 6.3. Diagrama de caso de uso de Gestión de inventario y facturación.



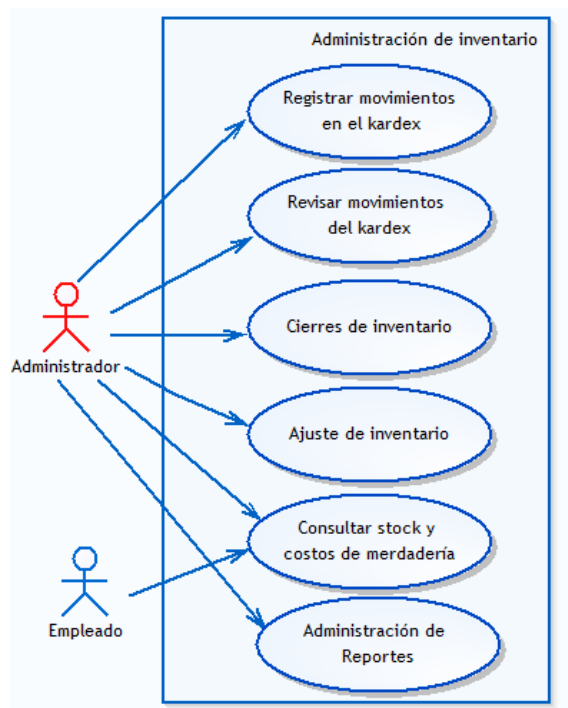
Especificación de casos de uso:

Tabla 6.1. Caso de uso: Iniciar Sesión

Caso de Uso: Iniciar de Sesión o Logging	
Actores:	Administrador, empleado
Descripción:	Permite ingresar al sistema y visualizar el menú según el perfil del usuario conectado.
Precondiciones:	El usuario debe estar registrado en el sistema y tener establecido un perfil de usuario.
Flujo Normal:	<ol style="list-style-type: none"> 1. El visitante acceder al formulario de validación de usuarios 2. El visitante digita su usuario y contraseña 3. Pulsar el botón Conectar.
Flujo Alternativo:	<ol style="list-style-type: none"> 2.1. El sistema comprueba la validez de los datos <ol style="list-style-type: none"> 2.1.1. Si los datos son correctos ingresa el usuario al sistema 2.1.2. Si son incorrectos se muestra la página de chequeo de seguridad.
Pos condiciones:	El usuario ingresa a la página principal con las opciones de menú al cual tiene acceso según su perfil.

▪ **Caso de uso: Administración de inventario**

Figura 6.4. Diagrama de caso de uso: Administración de inventario



Detalle del caso de uso: Registrar movimientos en kardex.

Tabla 6.2. Caso de uso: Registrar movimientos en kardex.

Caso de Uso: Registrar movimientos en kardex.
<p>Actores: Administrador</p>
<p>Descripción: El usuario accede al sistema para registra los movimientos de entrada y salida en el kardex de una manera manual.</p>
<p>Precondiciones: Tener artículos asignados a una bodega.</p>
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Abrir el listado de las kardex 2. Seleccionar el kardex. 3. Pulsar el botón Detalle. 4. Pulsar el botón Inventario inicial, ingresar mercadería o retirar mercadería. 5. Llenar los datos requeridos. 6. Pulsar el botón Guardar.
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 1.1 El usuario no tiene permiso para realizar la acción. 6.1 El sistema verifica la validez de los datos <ol style="list-style-type: none"> 6.1.1 Si los datos son correctos se almacenan y envía un mensaje indicando que se almacenado correctamente. 6.1.2 Si los datos son incorrectos no se almacena la información. 6.1.3 Si se generó alguna excepción los datos no se almacenan y se envía un mensaje con la excepción.
<p>Pos condiciones: Los datos de movimiento se reflejan en el kardex.</p>

Detalle del caso de uso: Revisar movimientos del kardex.

Tabla 6.3. Caso de uso: Revisar movimientos en kardex.

Caso de Uso: Revisar movimientos del kardex
<p>Actores: Administrador</p>
<p>Descripción: El usuario de sistema revisa detalladamente los movimientos que se han registrado en el kardex.</p>
<p>Precondiciones: Tener artículos asignados a una bodega.</p>

CONTINÚA

Tabla 6.3 (cont.). Caso de uso: Revisar movimientos del kardex.

<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Abrir el listado de las kardex 2. Seleccionar el kardex a revisar 3. Pulsar el botón Detalle.
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 1.1 El usuario no tiene permiso para realizar la acción. 3.1 No ha seleccionado ninguna kardex no hace nada. 3.2 No existen registros de movimientos.
<p>Pos condiciones:</p> <p>Los datos de los movimientos registrados se muestran a detalle.</p>

Detalle del caso de uso: Toma de inventario.

Tabla 6.4. Caso de uso: Toma de inventario.

Caso de Uso: Cierre de inventario
<p>Actores:</p> <p>Administrador</p>
<p>Descripción:</p> <p>El usuario accede al sistema para generar el cierre de inventario a fin de mes.</p>
<p>Precondiciones:</p> <p>Tener artículos asignados a una bodega.</p>
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Abrir el listado de cierres 2. Pulsar el botón Generar cierre 3. Llenar los datos requeridos 4. Pulsar el botón Guardar
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 4.1 El sistema verifica la validez de los datos <ol style="list-style-type: none"> 4.1.1 Si los datos son correctos se almacenan y envía un mensaje indicando que se almacenado correctamente. 4.1.2 Si los datos son incorrectos no se almacena la información. 4.1.3 Si se generó alguna excepción los datos no se almacenan y se envía un mensaje con la excepción.
<p>Pos condiciones:</p> <p>Los datos del cierre muestran en el listado.</p>

Detalle del caso de uso: Ajuste de inventario.

Tabla 6.5. Caso de uso: Ajuste de inventario.

Caso de Uso: Ajuste de inventario
Actores: Administrador
Precondiciones: Tener un cierre de inventario generado con estado del cierre <i>Generado</i> .
Flujo Normal: <ol style="list-style-type: none"> 1. Abrir el listado de cierres 2. Seleccionar un cierre 3. Pulsar el botón Ajustar inventario 4. Llenar los datos requeridos 5. Pulsar el botón Guardar
Flujo Alternativo: <ol style="list-style-type: none"> 3.1 No ha seleccionado ningún cierre no hace nada. 4.1 El sistema verifica la validez de los datos <ol style="list-style-type: none"> 4.1.1 Si los datos son correctos se almacenan y envía un mensaje indicando que se almacenado correctamente. 4.1.2 Si los datos son incorrectos no se almacena la información. 4.1.3 Si se generó alguna excepción los datos no se almacenan y se envía un mensaje con la excepción.
Pos condiciones: Los datos del ajuste se almacenan. presentar

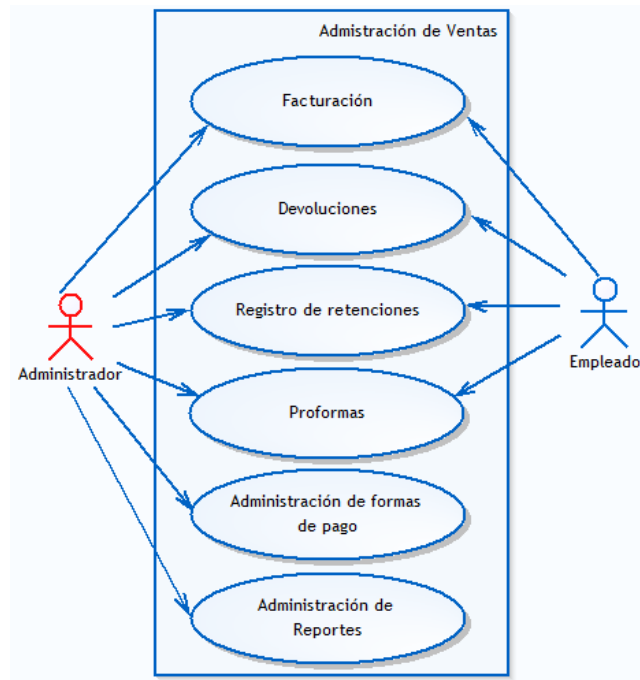
Detalle del caso de uso: Consultar stock y costos de mercadería.

Tabla 6.6. Caso de uso: Consultar stock y costos de mercadería

Caso de Uso: Consultar stock y costos de mercadería
Actores: Administrador, Empleado
Descripción: El usuario accede al sistema para consultar el stock y el costo de la mercadería existente en inventario
Precondiciones: Tener mercadería en inventario
Flujo Normal: <ol style="list-style-type: none"> 1. Abrir el listado de inventario
Flujo Alternativo: <ol style="list-style-type: none"> 1.1 El usuario no tiene permiso para realizar la acción. 1.2 La conexión a la base de datos se interrumpió.
Pos condiciones: Los datos del inventario se muestran detalladamente.

- **Caso de uso: Administración de ventas**

Figura 6.5. Diagrama de caso de uso: Administración de ventas



Especificaciones de casos de uso:

Detalle del caso de uso: Facturación

Tabla 6.7. Caso de uso: Facturación.

Caso de Uso: Facturación	
Actores:	Administrador, empleado
Descripción:	El usuario accede al sistema para facturar de productos y/o servicios.
Precondiciones:	Tener clientes registrados Tener artículos y servicios registrados y en stock Tener formas de pago registradas Tener llenos los parámetros de configuración
Flujo Normal:	<ol style="list-style-type: none"> 1. Pulsar la opción Facturación 2. Pulsar el botón Nuevo 3. Pulsar el botón Seleccionar cliente 4. Marcar el cliente

CONTINÚA

Tabla 6.7 (cont.). Caso de uso: Facturación

<p>5. Pulsar el botón Seleccionar 6. Pulsar el botón Añadir productos 7. Marcar los artículos y/o servicios solicitados 8. Pulsar el botón Seleccionar 9. Asignar valores a las formas de pago 10. Pulsar el botón Guardar</p>
<p>Flujo Alternativo: 3.1 El cliente no está registrado 3.1.1 Si se generó alguna excepción no se muestra la información 5.1 No ha seleccionado ningún cliente no hace nada 6.1 No existe el producto solicitado 6.1.1 Si se generó alguna excepción no se muestra la información 8.1 No ha seleccionado ningún producto no hace nada 9.1 No se ha registrado las formas de pago 9.1.1 Los valores asignados a las formas de pago son incorrectos 10.1 El sistema verifica la validez de los datos 10.1.1 Si los datos son correctos se almacenan, se envía un mensaje indicando se ha almacenado correctamente y se genera la factura en formato pdf. 10.1.2 Si los datos son incorrectos no se almacenan 10.1.3 Si se generó alguna excepción los datos no se almacenan y se envía un mensaje indicando la excepción generada.</p>
<p>Excepciones: La conexión con la base de datos no está establecida o se interrumpió. No tiene permisos para realizar la acción.</p>
<p>Pos condiciones: La factura se visualiza en formato pdf y la información generada de la venta se refleja en el listado de facturas. Se descarga el producto de inventario Se registra el movimiento en contabilidad</p>

Detalle del caso de uso: Devoluciones

Tabla 6.8. Caso de uso: Devoluciones.

Caso de Uso: Devoluciones
<p>Actores: Administrador, empleado</p>
<p>Descripción: El usuario accede al sistema para realizar la devolución de productos facturados.</p>
<p>Precondiciones: Tener facturas generadas Que la factura este en un estado vigente y no haya sido devuelta. Que la factura este dentro del tiempo máximo de devolución</p>

CONTINÚA

Tabla 6.8 (cont.). Caso de uso: Devoluciones

<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Seleccionar la opción Listado de devoluciones 2. Pulsar el botón Nuevo 3. Pulsar el botón Seleccionar factura 4. Marcar la factura 5. Pulsar el botón Seleccionar 6. Pulsar el botón Añadir productos 7. Marcar los artículos y/o servicios devueltos 8. Pulsar el botón Seleccionar 9. Asignar cantidades a devolver 10. Pulsar el botón Guardar
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 10.1 El sistema verifica la validez de los datos <ol style="list-style-type: none"> 10.1.1 Si los datos son correctos se almacenan, se envía un mensaje indicando se han almacenado correctamente y se genera el documento de respaldo de la devolución en formato pdf. 10.1.2 Si los datos son incorrectos no se almacenan 10.1.3 Si se generó alguna excepción, los datos no se almacenan y se envía un mensaje indicando la excepción generada.
<p>Excepciones:</p> <p>La conexión con la base de datos no está establecida o se interrumpió. No tiene permisos para realizar la acción.</p>
<p>Pos condiciones:</p> <p>Se genera el documento de respaldo en formato pdf y la información generada de la devolución se refleja en el listado de devoluciones. Se carga el producto a inventario Se registra el movimiento en contabilidad</p>

Detalle del caso de uso: Registro de retenciones

Tabla 6.9. Caso de uso: Registro de retenciones.

Caso de Uso: Registro de retenciones
<p>Actores:</p> <p>Administrador, empleado</p>
<p>Descripción:</p> <p>El usuario accede al sistema para ingresar los datos de la retención aplicada a la factura emitida por la empresa.</p>
<p>Precondiciones:</p> <p>Tener facturas generadas Que la factura este en un estado vigente</p>
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Seleccionar la opción Listado de retenciones

CONTINÚA

Tabla 6.9 (cont.). Caso de uso: Registro de retenciones

<ol style="list-style-type: none"> 2. Pulsar el botón Nuevo 3. Pulsar el botón Seleccionar factura 4. Marcar la factura 5. Pulsar el botón Seleccionar 6. Pulsar el botón Añadir impuesto 7. Llenar los campos requeridos 8. Pulsar el botón Guardar
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 8.1 El sistema verifica la validez de los datos <ol style="list-style-type: none"> 8.1.1 Si los datos son correctos se almacenan, se envía un mensaje indicando que se han almacenado correctamente. 8.1.2 Si los datos son incorrectos no se almacenan 8.1.3 Si se generó alguna excepción, los datos no se almacenan y se envía un mensaje indicando la excepción generada.
<p>Excepciones:</p> <p>La conexión con la base de datos no está establecida o se interrumpió. No tiene permisos para realizar la acción.</p>
<p>Pos condiciones:</p> <p>La información de la retención ingresada se refleja a detalle en el listado de las retenciones. Se registra el movimiento en contabilidad</p>

Detalle del caso de uso: Proformas

Tabla 6.10. Caso de uso: Proformas.

Caso de Uso: Proformas
<p>Actores:</p> <p>Administrador, empleado</p>
<p>Descripción:</p> <p>El usuario accede al sistema para generar proformas de productos y/o servicios</p>
<p>Precondiciones:</p> <p>Tener clientes registrados Tener artículos y servicios registrados Tener llenos los parámetros de configuración</p>
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Pulsar la opción Proformas 2. Pulsar el botón Nuevo 3. Pulsar el botón Seleccionar cliente 4. Marcar el cliente 5. Pulsar el botón Seleccionar 6. Pulsar el botón Añadir productos

CONTINÚA

Tabla 6.10 (cont.). Caso de uso: Proformas

<p>7. Marcar los artículos y/o servicios solicitados 8. Pulsar el botón Seleccionar 9. Asignar valores a las formas de pago 10. Pulsar el botón Guardar</p>
<p>Flujo Alternativo: 3.1 El cliente no está registrado 3.1.1 Si se generó alguna excepción no se muestra la información 5.1 No ha seleccionado ningún cliente no hace nada 6.1 No existe el producto solicitado 6.1.1 Si se generó alguna excepción no se muestra la información 8.1 No ha seleccionado ningún producto no hace nada 9.1 No se ha registrado las formas de pago 9.1.1 Los valores asignados a las formas de pago son incorrectos 10.1 El sistema verifica la validez de los datos 10.1.1 Si los datos son correctos se almacenan, se envía un mensaje indicando se ha almacenado correctamente y se genera la proforma en formato pdf. 10.1.2 Si los datos son incorrectos no se almacenan. 10.1.3 Si se generó alguna excepción los datos no se almacenan y se envía un mensaje indicando la excepción generada.</p>
<p>Excepciones: La conexión con la base de datos no está establecida o se interrumpió. No tiene permisos para realizar la acción.</p>
<p>Pos condiciones: La proforma se visualiza en formato pdf , la información generada de la proforma se refleja en el listado de proformas.</p>

Detalle del caso de uso: Administración de formas de pago

Tabla 6.11. Caso de uso: Administración de formas de pago.

Caso de Uso: Administración de formas de pago
<p>Actores: Administrador</p>
<p>Descripción: El usuario accede al sistema para ingresar una forma de pago.</p>
<p>Flujo Normal: 1. Abrir el listado de Formas de pago 2. Pulsar el botón Nuevo. 3. Llenar los datos requeridos. 4. Pulsar el botón Guardar.</p>
<p>Flujo Alternativo: 1.1 El usuario no tiene permiso para realizar la acción.</p>

CONTINÚA

Tabla 6.11 (cont.). Caso de uso: Administración de formas de pago

<p>4.1 El sistema verifica la validez de los datos</p> <p>4.1.1 Si los datos son correctos se almacenan, se envía un mensaje indicando que se almacenado correctamente.</p> <p>4.1.2 Si los datos son incorrectos no se almacenan</p> <p>4.1.3 Si se generó alguna excepción los datos no se almacenan y se envía un mensaje indicando la excepción generada.</p>
<p>Excepciones: La conexión con la base de datos no está establecida o se interrumpió.</p>
<p>Pos condiciones: Los datos almacenados se reflejan en el listado de formas de pago.</p>

Detalle del caso de uso: Administración de reportes

Tabla 6.12. Caso de uso: Administración de reportes

Caso de Uso: Administración de reportes
<p>Actores: Administrador</p>
<p>Descripción: El usuario accede al sistema para generar reportes generales e individuales de ventas e inventario.</p>
<p>Flujo Normal:</p> <ol style="list-style-type: none"> 1. Seleccionar la opción de Reportes 2. Seleccionar el reporte 3. [Ingresar los parámetros del reporte] 4. Pulsar el botón Generar
<p>Flujo Alternativo:</p> <ol style="list-style-type: none"> 4.1 No hay datos que mostrar. 4.2 Se rompió el vinculo del reporte
<p>Excepciones: La conexión con la base de datos no está establecida o se interrumpió. La conexión con el servidor de reportes no está establecida o se interrumpió.</p>
<p>Pos condiciones: Se visualiza el reporte generado en formato pdf.</p>

Prototipo de interfaz de usuario

Un prototipo de interfaz de usuario es una representación parcial de la interfaz de usuario que tendrá el software, muestra la forma en que el usuario interactuará con él.

A continuación se presenta los prototipos que servirán como referencia para los demás ya que todos tienen un formato similar.

Para diseñar los prototipos de interfaz se utilizó Microsoft Office Visio 2007.

- **Prototipo de la interfaz del caso de uso Iniciar Sesión**

Figura 6.6. Página para Iniciar sesión

Validación de usuarios

Usuario:

Contraseña:

- **Prototipo de la interfaz del detalle de caso de uso Registrar movimientos en kardex**

Figura 6.7. Formulario para Seleccionar el tipo de movimiento de la kardex

Tarjeta kardex

Logo

Datos del artículo

Código:

Descripción:

Grupo:

Marca:

Datos del kardex

Bodega:

Método:

Existencias:

Movimientos

Fecha	Concepto	Cant	V.U. Ent	V.T. Ent	Cant	V.U. Sal	V.T. Sal	Cant	V.U. Ext	V.T. Ext

Figura 6.8. Formulario para Registrar movimiento de la kardex

Tarjeta kardex

Logo

Datos del movimiento

Fecha:

Concepto:

Cantidad:

Valor unitario:

Valor total:

Observación:

Guardar Cancelar Cerrar

- Prototipo de la interfaz del detalle de caso de uso Facturación

Figura 6.9. Formulario para Facturación

Factura

Logo

Datos del cliente

Código:

CI/RUC:

Nombres:

Dirección:

Teléfono:

Datos del comprobante

N° Factura:

Fecha emisión:

Vendedor:

Formas de pago:

Detalle

Cantidad	Código	Bodega	Descripción	Unidad	Prec. Unt	Total

Observación:

Subtotal:

IVA:

Total:

Guardar Cancelar Cerrar

- Prototipo de la interfaz del detalle de caso de uso Devoluciones

Figura 6.10. Formulario para Devoluciones

The 'Devolución' form is structured as follows:

- Logo:** A text input field.
- Datos del cliente:** Includes 'Factura' (text input) and 'Cliente' (text input). A 'Seleccionar factura' button is next to the Factura field.
- Datos del comprobante:** Includes 'Nº Factura:' (text input) and 'Fecha emisión:' (text input).
- Detalle:** A table with columns: Cantidad, Código, Bodega, Descripción, Unidad, Prec. Unt, Total. Above the table are buttons for 'Añadir productos' and 'Eliminar producto'.
- Observación:** A large text area.
- Summary:** Fields for 'Subtotal:', 'IVA:', and 'Total:'.
- Buttons:** 'Guardar', 'Cancelar', and 'Cerrar' at the bottom.

- Prototipo de la interfaz del detalle de caso de uso Registro de retenciones

Figura 6.11. Formulario para Registro de retenciones

The 'Retenciones' form is structured as follows:

- Logo:** A text input field.
- Datos Generales:** Includes 'Factura:' (text input) and 'Número de retención:' (text input). A 'Seleccionar factura' button is next to the Factura field. To the right are 'Total:' (text input) and 'Adjunto:' (checkbox).
- Impuestos:** A table with columns: Ejercicio fiscal, Base imponible, Impuesto, % de Retención, Valor retenido. Above the table are buttons for 'Añadir impuesto' and 'Eliminar impuesto'.
- Buttons:** 'Guardar', 'Cancelar', and 'Cerrar' at the bottom.

- Prototipo de la interfaz del detalle de caso de uso Proformas

Figura 6.12. Formulario para Generar proformas

The screenshot shows a window titled "Proforma" with a blue title bar. The window contains the following elements:

- Logo:** A large empty rectangular box at the top.
- Datos del cliente:** A section with six input fields: Código, CI/RUC, Apellido, Nombre, Dirección, and Teléfono.
- Datos proforma:** A section with four input fields: N° Proforma, Fecha emision, Fecha vencimiento, and Vendedor.
- Detalle:** A section containing two buttons: "Añadir producto" and "Eliminar producto". Below them is a table with five columns: Cantidad, Descripción, Unidad, Prec. Unt, and Total. The table has four empty rows.
- Observación:** A single-line text input field.
- Summary:** Three input fields labeled Subtotal, IVA, and Total.
- Buttons:** "Guardar" and "Cancelar" at the bottom left, and "Cerrar" at the bottom right.

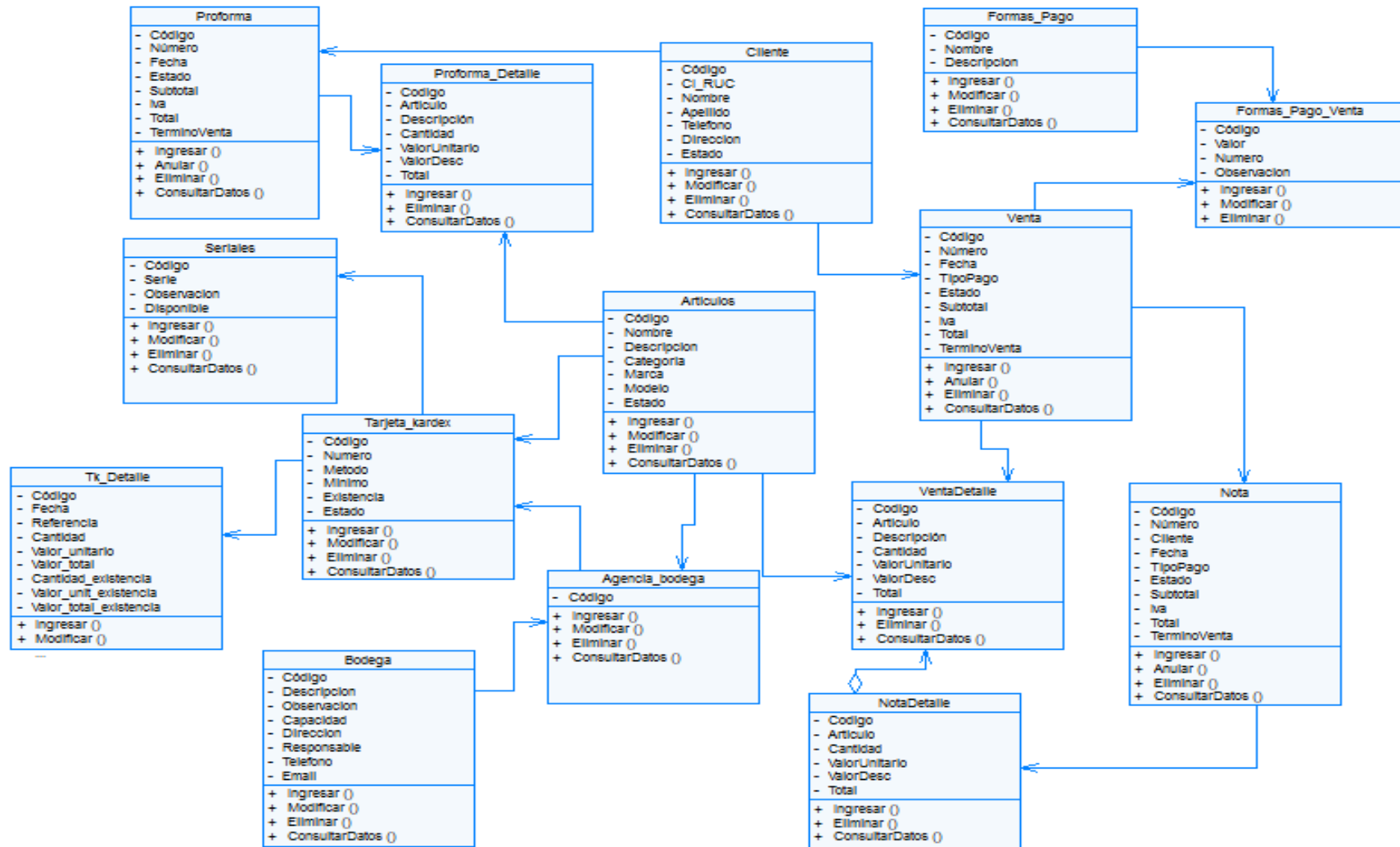
Diagramas de clases

Los diagramas de clases se usan para modelar gráficamente la vista estática del software. Describen los tipos de objetos que son importantes para modelar un sistema y cómo se relacionan. Ver Figura 6.13

Diagrama de secuencia

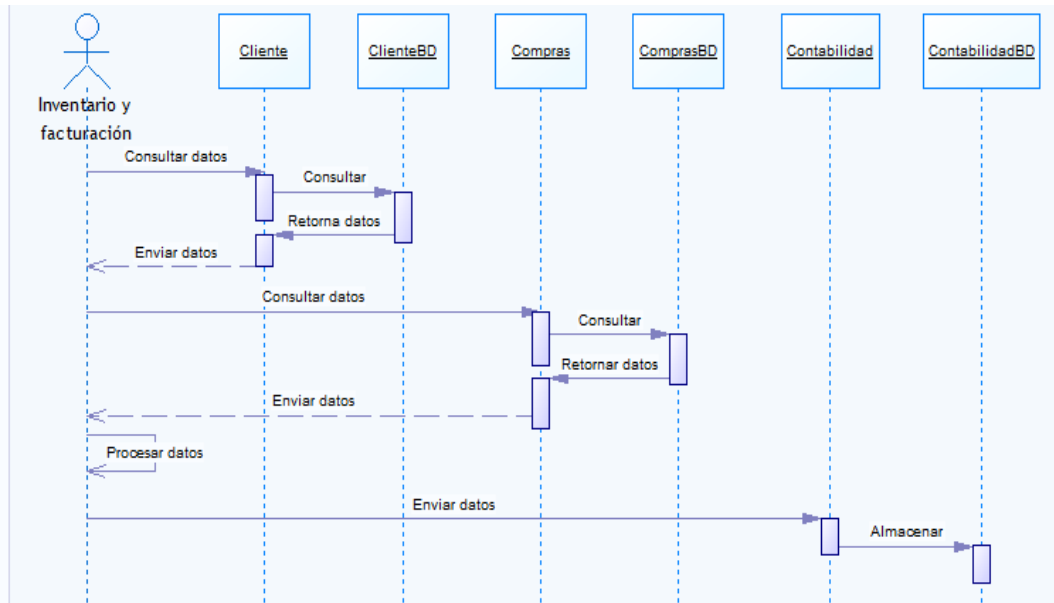
El Diagrama de Secuencia es uno de los diagramas más efectivos para modelar interacción entre objetos en un sistema. Muestra la secuencia de los mensajes entre objetos durante un escenario concreto Ver Página 76.

Figura 6.13. Diagrama de clases del módulo de gestión de inventario y facturación



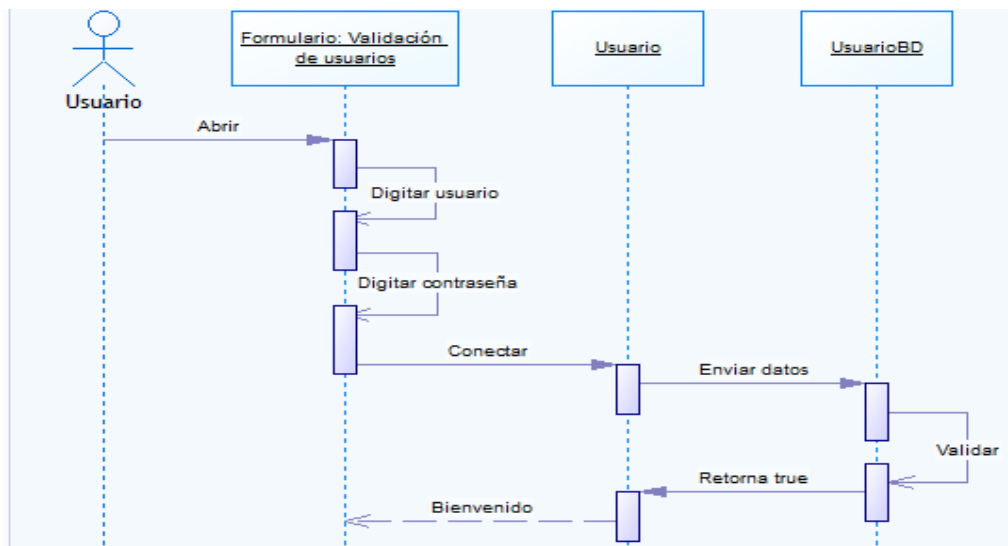
- Diagrama de secuencia de Interacción del Módulo de gestión de inventario y facturación y los módulos de clientes, compras y contabilidad.

Figura 6.14. Diagrama de secuencia Interacción entre módulos



- Diagrama de secuencia de Iniciar Sesión

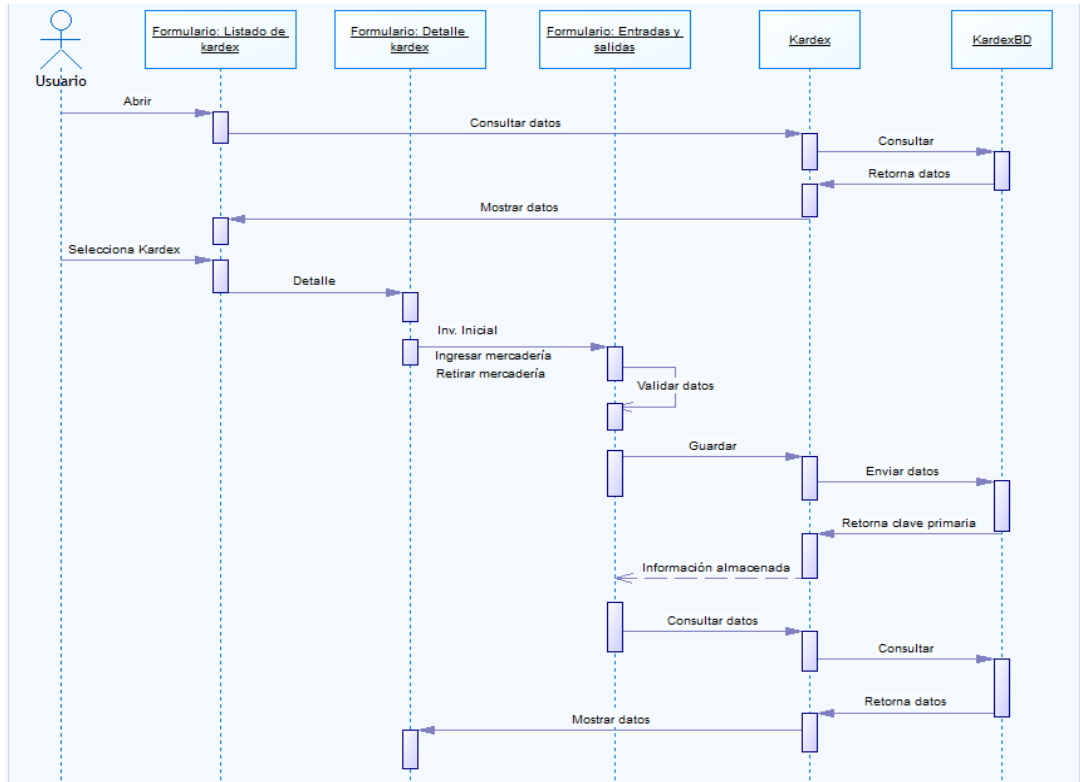
Figura 6.15. Diagrama de secuencia Iniciar sesión



- Diagrama de secuencia de Administración de inventario

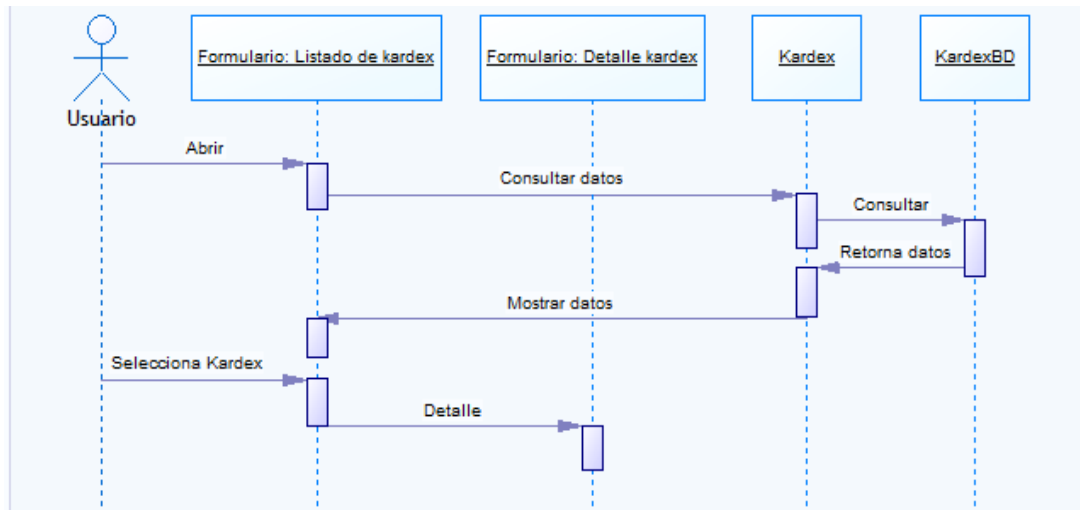
Registrar movimientos en kardex.

Figura 6.16. Diagrama de secuencia Registrar movimientos en kardex



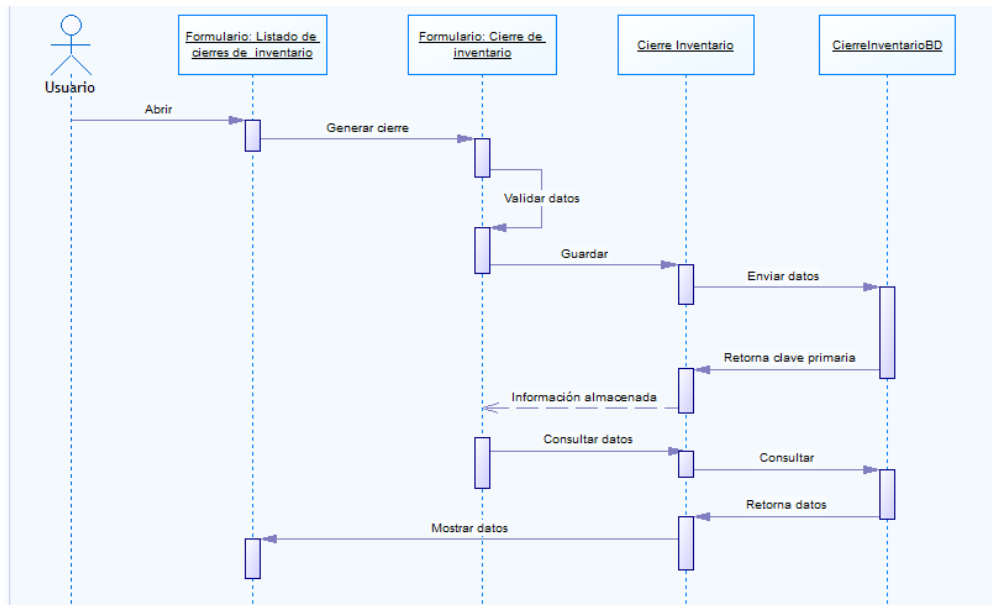
Revisar movimientos del kardex

Figura 6.17. Diagrama de secuencia Revisar movimientos del kardex



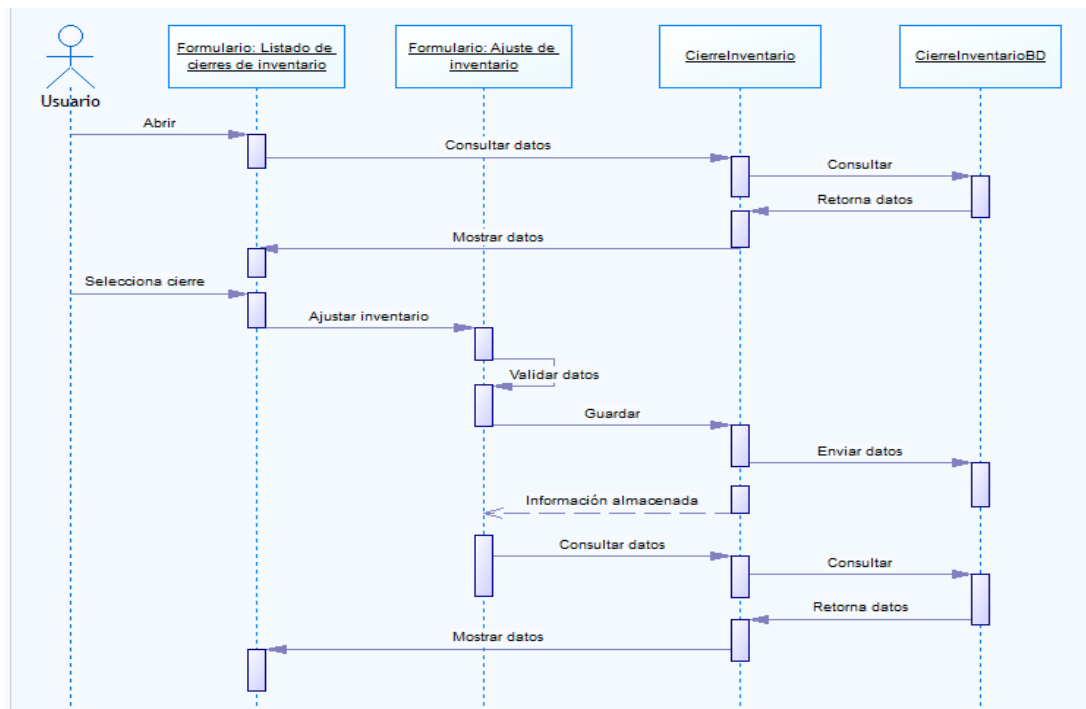
Toma de inventario.

Figura 6.18. Diagrama de secuencia Toma de inventario



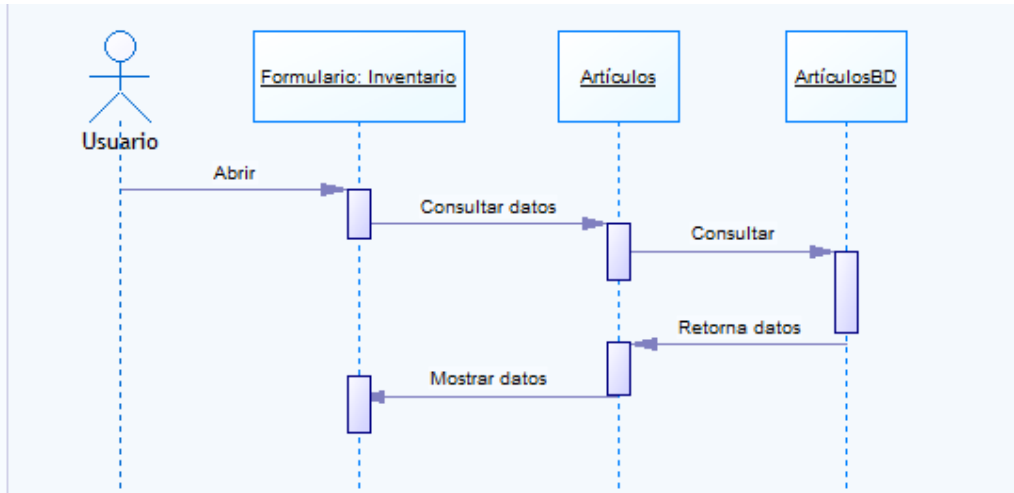
Ajuste de inventario.

Figura 6.19. Diagrama de secuencia Ajuste de inventario



Consultar stock y costos de mercadería.

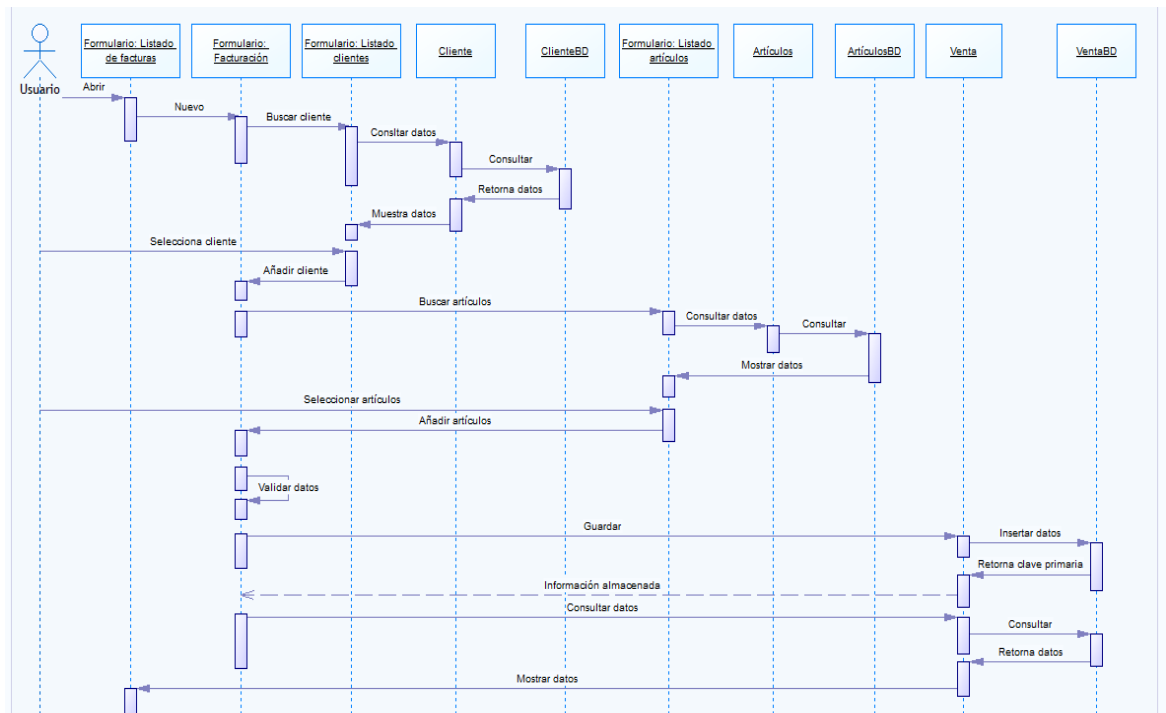
Figura 6.20. Diagrama de secuencia Consultar stock y costos de mercadería



▪ Diagrama de secuencia de Administración de ventas

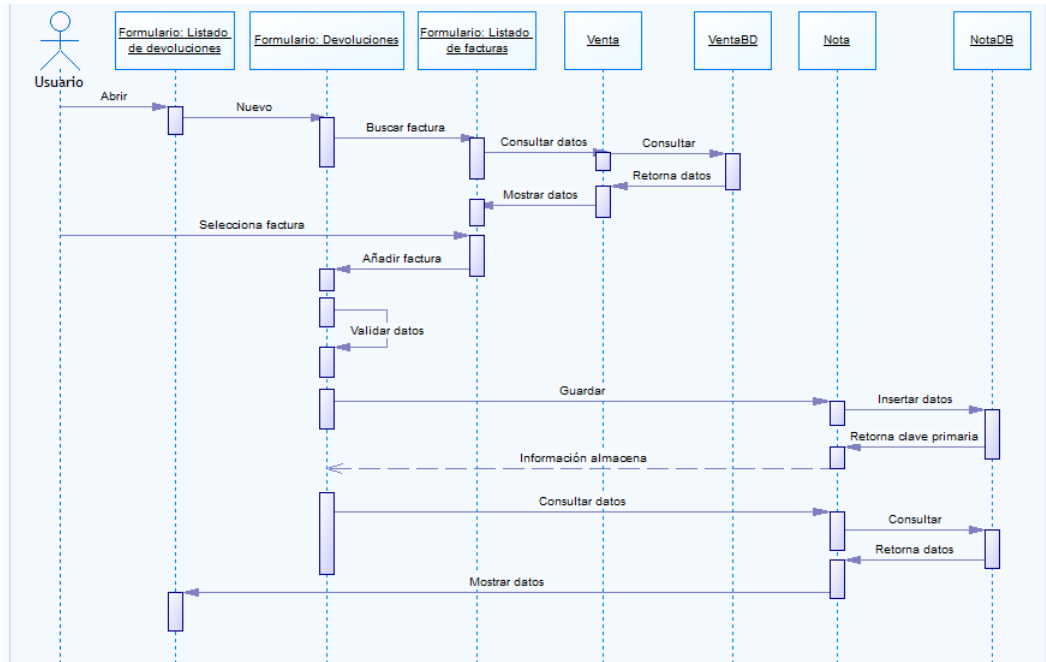
Facturación

Figura 6.21. Diagrama de secuencia Facturación



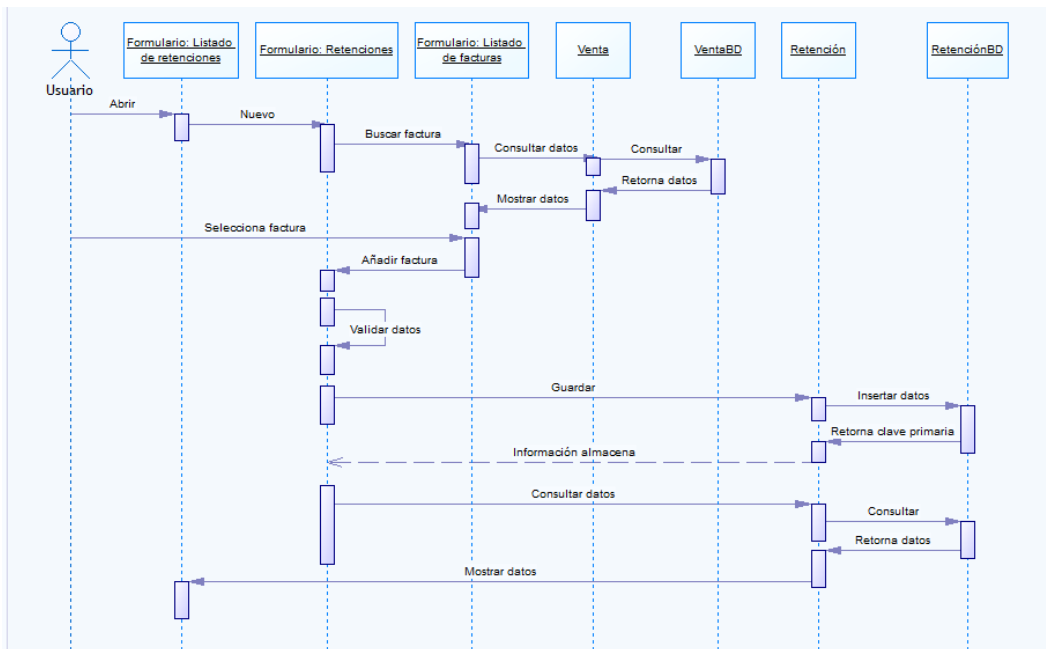
Devoluciones

Figura 6.22. Diagrama de secuencia Devoluciones



Registro de retenciones

Figura 6.23. Diagrama de secuencia Registro de retenciones



Administración de formas de pago

Figura 6.24. Diagrama de secuencia Administración de formas de pago

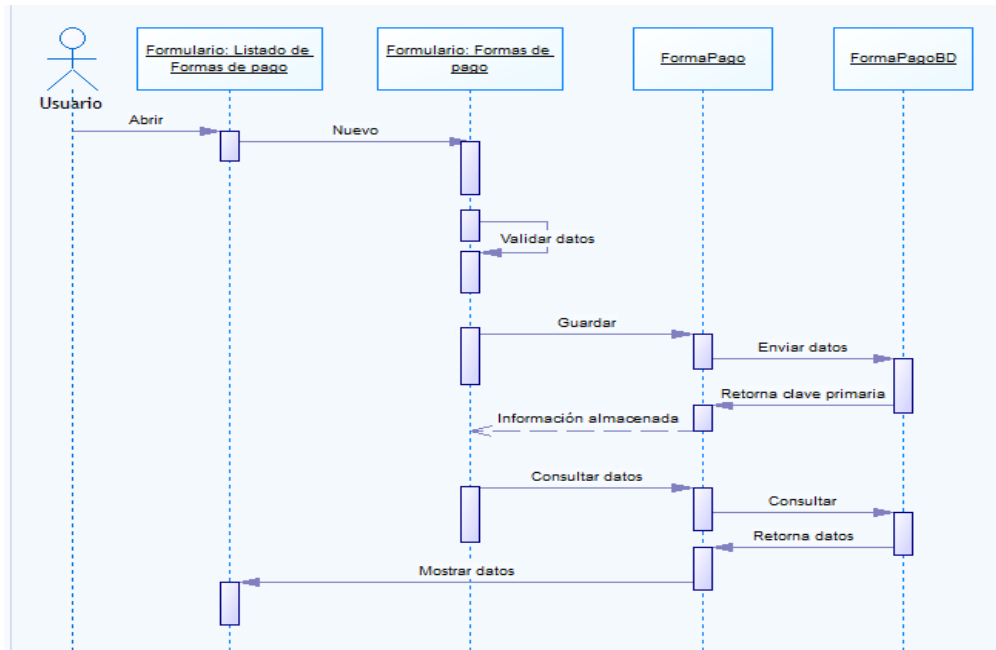
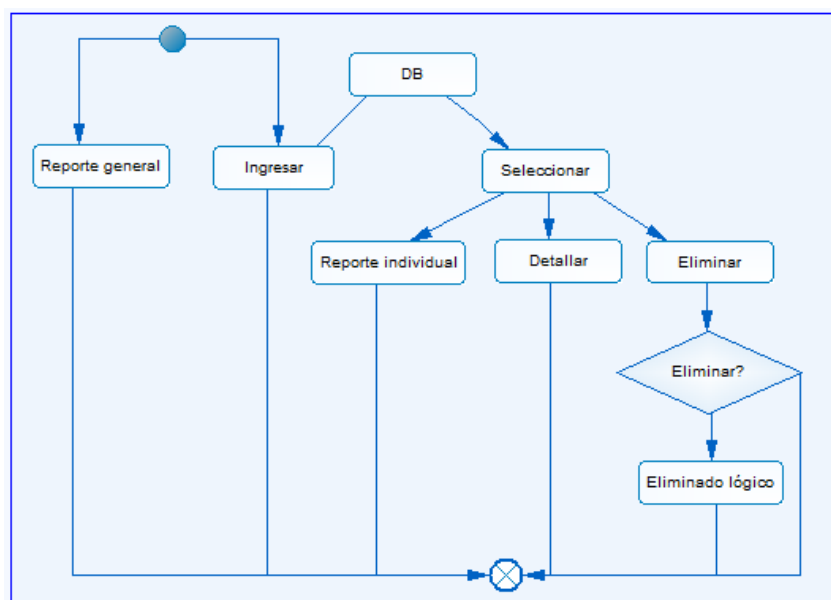


Diagrama de actividades

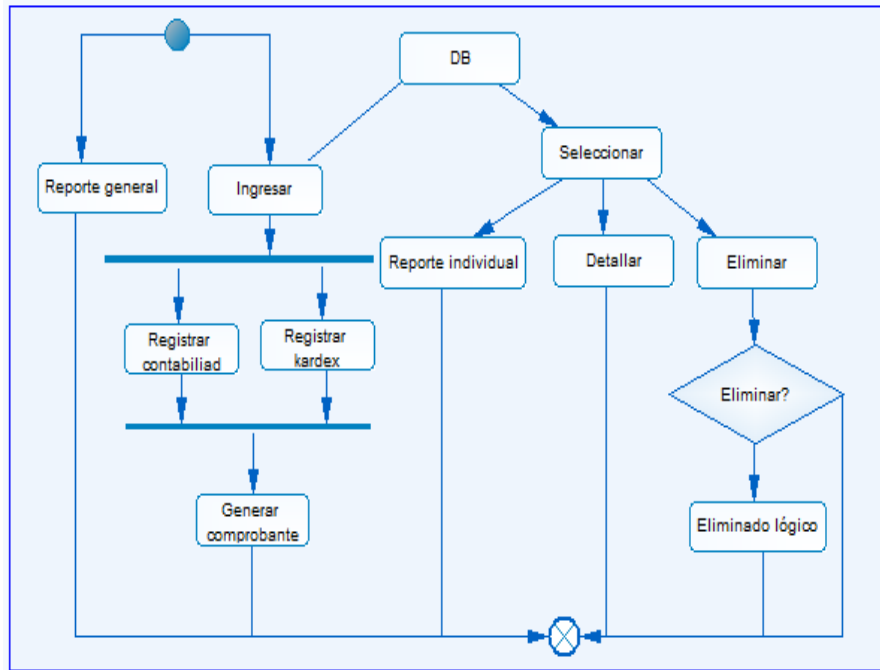
- Diagrama de actividades general de la Administración de inventario

Figura 6.25. Diagrama de actividades Administración de inventario



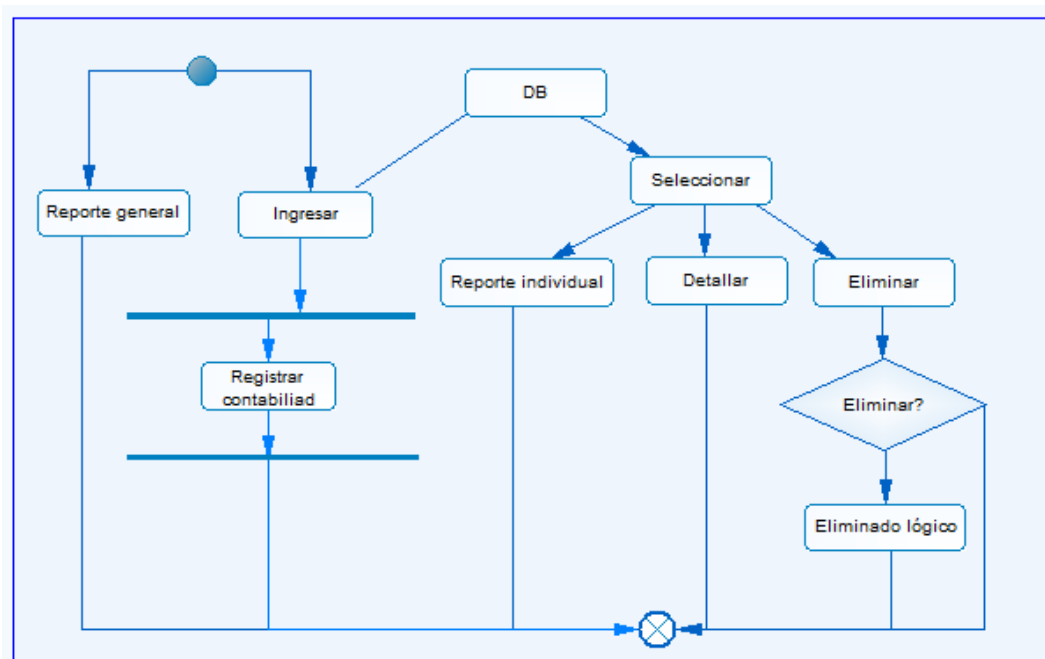
- **Diagrama de actividades de Facturación y devoluciones**

Figura 6.26. Diagrama de actividades Facturación y devoluciones



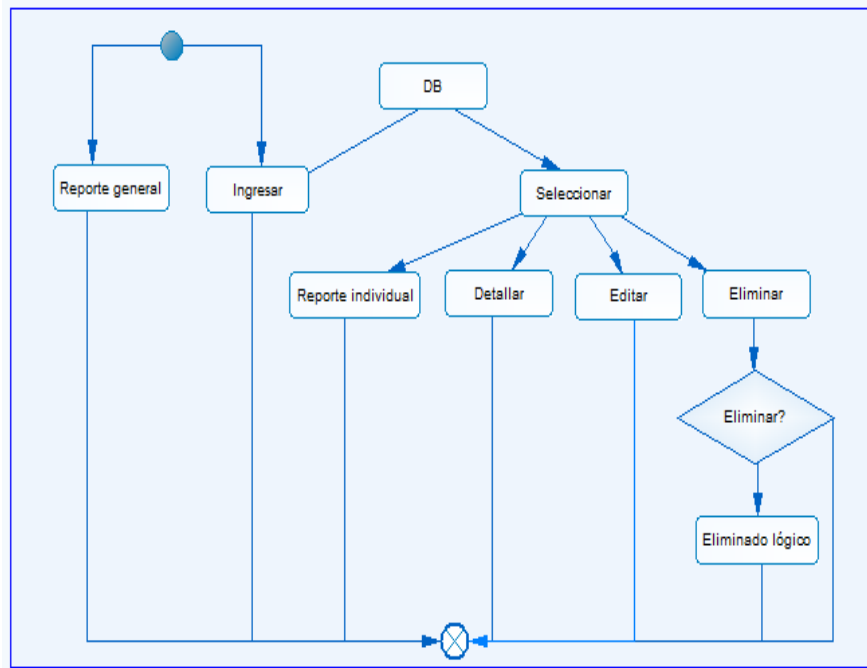
- **Diagrama de actividades de Registro de retenciones**

Figura 6.27. Diagrama de actividades Registro de retenciones



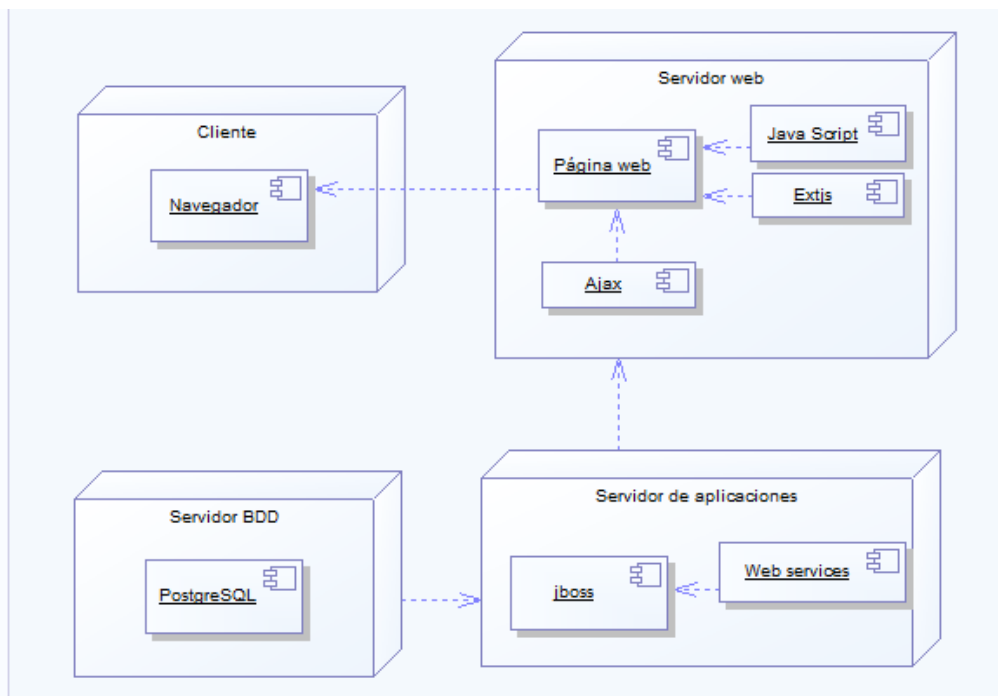
- **Diagrama de actividades de proformas y formas de pago**

Figura 6.28. Diagrama de actividades Proformas y formas de pago



6.8.2. Diagrama de despliegue

Figura 6.29. Diagrama de despliegue del módulo de gestión de inventario y facturación



6.8.3. Diseño del sistema

6.8.3.1. Arquitectura de software

La arquitectura de software es el diseño del más alto nivel. Consiste en definir cuáles serán los componentes que formarán el software. La arquitectura debe favorecer el cumplimiento de los requerimientos funcionales y no funcionales especificados para el producto.

Las cualidades que debe tener la arquitectura son:

- Sencillez. Que sea fácil de comprender y de implementar.
- Extensión. La posibilidad de agregar nuevos componentes.
- Cambio. Que los cambios en los requerimientos no afecten mucho a la arquitectura.

El modelo de capas, es la base de la arquitectura de un sistema cuando es posible estructurar la solución en grupos de tareas, donde cada grupo tiene un nivel de abstracción particular.

Una *capa* es una abstracción que toma el resultado de la capa inferior, efectúa su función y entrega ese resultado a la capa superior. La regla en este modelo es que cada capa se comunica solamente con las capas adyacentes.

Para definir la arquitectura se usarán las siguientes capas:

- Capa de Presentación. En esta capa se ponen los elementos con los que interactúa directamente el usuario: las ventanas, páginas, informes, etc.
- Capa Lógica de la aplicación. Son los elementos que implementan las reglas del negocio y la lógica de la aplicación.
- Capa de Almacenamiento. Contiene los elementos que guardan la información para asegurar su persistencia tales como bases de datos o archivos.

Arquitectura de software SOA.- Una arquitectura orientada a servicios es una estrategia donde las aplicaciones hacen uso (o se basan) en servicios disponibles en una red. Siendo una manera de compartir funciones (típicamente de negocios) en una manera flexible y extendida.

Niveles y capas de la SOA.

- Nivel de datos
 - Capa de datos
- Nivel de servicios
 - Capa de acceso a datos
 - Capa de lógica de negocio
 - Capa de servicios
- Nivel de clientes.
 - Capa de presentación

Capa.- Referencia a las distintas "partes" en que una aplicación se divide desde un punto de vista lógico.

Nivel.- Es la forma en que las capas lógicas se encuentran distribuidas de forma física.

Ventajas de SOA:

- Respuesta rápida a nuevas necesidades de negocio

Al tener todas las capacidades del negocio representadas en servicios, cuando se necesita una nueva interacción, simplemente se pueden utilizar los servicios existentes por medio de una nueva colaboración, limitando los desarrollos a capacidades nuevas. Se asume que la creación de una colaboración de servicios es mucho más sencilla que el desarrollo de las capacidades totales de los servicios e, incluso, es más simple que hacer interfaces propietarias para la colaboración. La implementación de SOA debe ser adecuada para que esta asunción sea realidad.

- Reducción del costo de desarrollo de IT

El hecho de que se necesite menos desarrollo de integración, reduce los costos de IT que estarían destinados a este fin. A su vez, reduce el nivel de acoplamiento, por lo que un cambio en un servicio no afectará a los consumidores. Mejora la definición de los roles de desarrollo. Al implementar cada función de negocio como un servicio diferente, permite delimitar mejor las responsabilidades de cada programador. Facilita el testeo. El bajo acoplamiento y la granularidad permiten un testeo más específico de las funciones. Mejora el mantenibilidad. Favorece la reutilización y mejora la productividad. Favorece el desarrollo en paralelo.

- Capacidad de integrar a clientes y socios

Una gran ventaja de SOA es que está estandarizado. Existen varias especificaciones de SOA, lo cual permite que distintas empresas puedan publicar parte de los servicios de su negocio para que clientes, proveedores y socios puedan acceder a ellos. En particular, Web Services es adecuado para esto, ya que está basado en tecnologías ubicuas (HTTP, XML, etc.). Existen además muchas herramientas útiles que lo soportan.

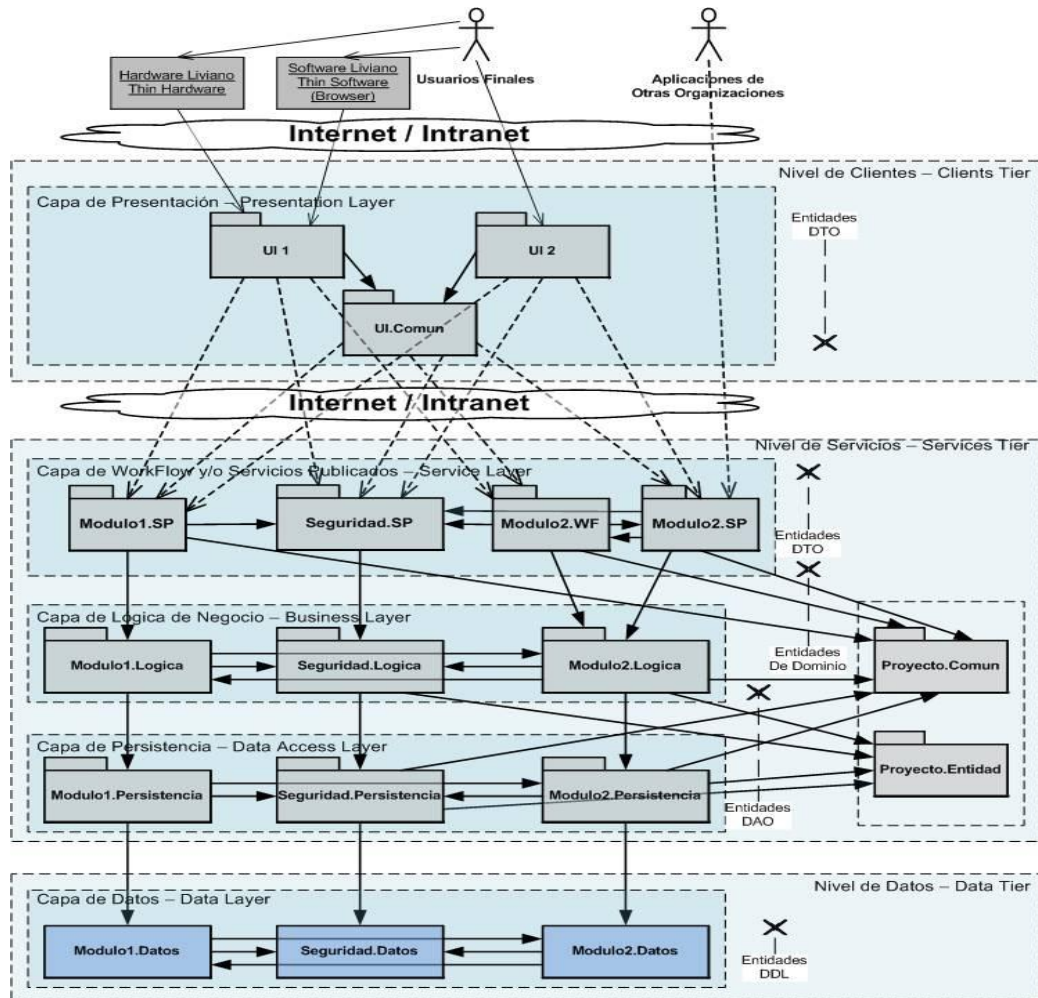
- Capacidad de generar nuevos modelos de negocios

Con la aparición de herramientas de modelado de negocio e integración de servicios, se pueden analizar nuevos modelos de negocios, ver cómo los servicios actuales los soportan y ver qué necesidades deben completarse.

- Alinear objetivos de IT a objetivos de negocio

La agilidad para responder a nuevas necesidades de negocio hace más sencillo mantener alineados los objetivos de IT con los del negocio. Se logra un mapeo más directo entre los procesos de negocio y los sistemas. La composición de servicios permite replicar los procesos del negocio del mundo físico en el mundo virtual, de un modo más directo y natural.

Figura 6.30. Niveles y capas de la SOA



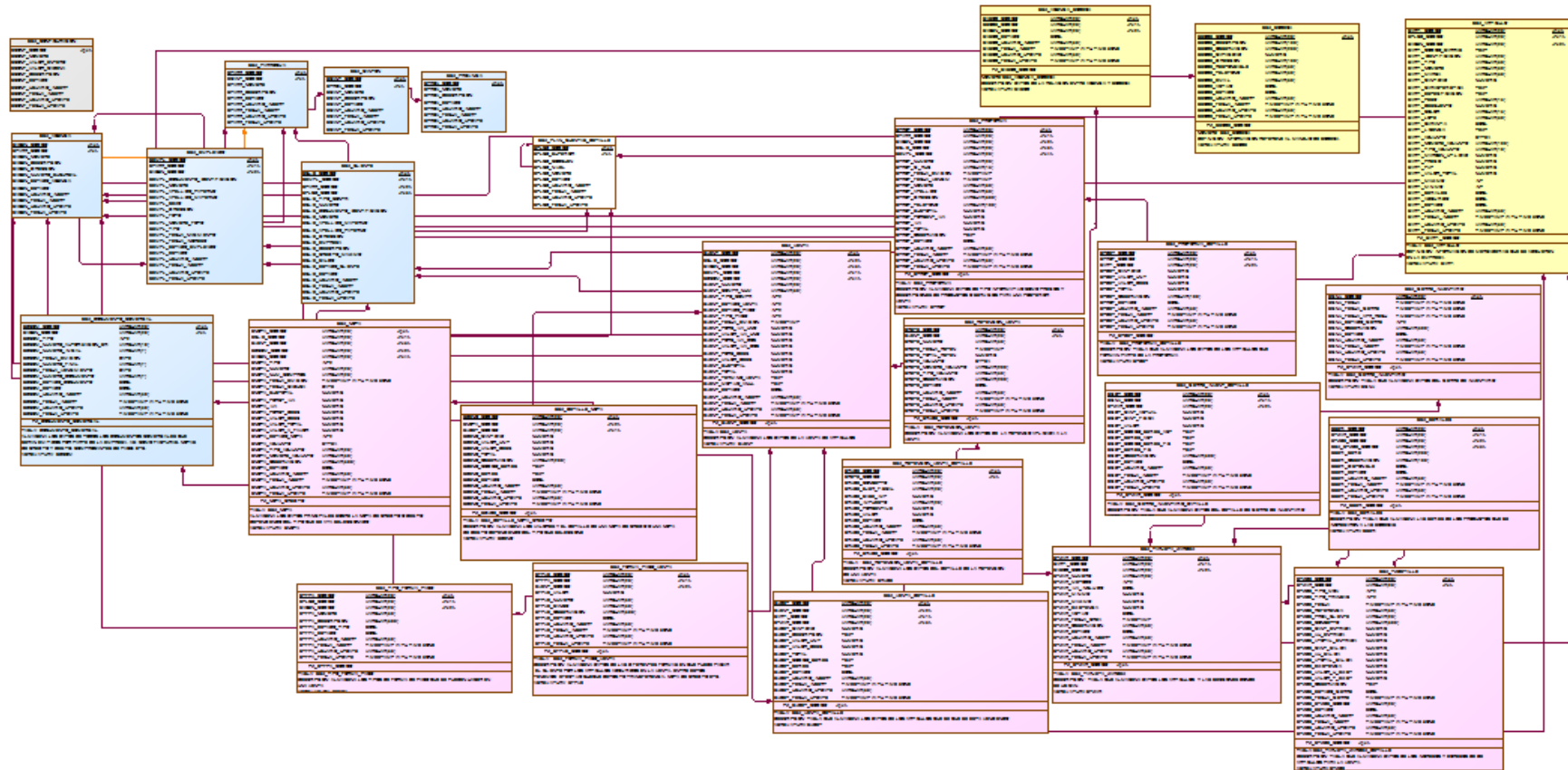
6.8.3.2. Diseño de la base de datos

El módulo de inventario y facturación forma parte del sistema Administrativo Contable de la empresa Besixplus Cia. Ltda, y para diseñar la base de datos se ha tomado en cuenta tablas del modulo de contabilidad, clientes y compras. Ver Figura 6.31.

Descripción de tablas

	Módulo de gestión de inventario y facturación
	Módulo de compras y proveedores
	Módulo de clientes y seguridades
	Módulo de contabilidad
	Parámetros de configuración

Figura 6.31. Diseño de base de datos Módulo de Gestión de inventario y facturación



6.8.3.3. Diccionario de datos

Tabla 6.13. Descripción de la tabla BSX_BODEGA

Tabla: BSX_BODEGA			
Nombre	Comentario	Tipo de dato	Clave
Bbode_codigo	Clave primaria de la tabla bsx_bodega	varchar(20)	pk
Bbode_descripcion	Descripción de la bodega	varchar(100)	
Bbode_observacion	Observación sobre la bodega	varchar(200)	
Bbode_capacidad	Capacidad de la bodega en metros cuadrados	numeric	
Bbode_direccion	Ubicación de la bodega	varchar(100)	
Bbode_responsable	Responsable de la bodega.	varchar(50)	
Bbode_telefono	Teléfono de la bodega	varchar(50)	
Bbode_email	e-mail de la bodega	varchar(50)	
Bbode_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool	
Bbode_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Bbode_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Bbode_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Bbode_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.14. Descripción de la tabla BSX_AGENCIA_BODEGA

Tabla: BSX_AGENCIA_BODEGA			
Nombre	Comentario	Tipo de dato	Clave
Babod_codigo	Clave primaria de la tabla bsx_agencia_bodega	varchar(20)	pk

CONTINÚA

Tabla 6.14 (cont.). Descripción de la tabla BSX_AGENCIA_BODEGA

Bbode_codigo	Clave primaria de la tabla bsx_bodega	varchar(20)	fk
Bagen_codigo	Clave primaria de la tabla bsx_agencia	varchar(20)	fk
Babod_estado	"estado del registro.		bool
Babod_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Babod_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Babod_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Babod_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.15. Descripción de la tabla BSX_TARJETA_KARDEX

Tabla: BSX_TARJETA_KARDEX			
Nombre	Comentario	Tipo de dato	Clave
Btkar_codigo	Clave primaria de la tabla bsx_tarjeta_kardex	varchar(20)	pk
Barti_codigo	Clave primaria de la tabla bsx_articulo	varchar(20)	fk
Babod_codigo	Clave primaria de la tabla bsx_agencia_bodega	varchar(20)	fk
Btkar_numero	Número de la tarjeta kardex	numeric	
Btkar_metodo	Método de la tarjeta kardex 0 lifo (ultimo en entrar primero en salir) 1 fifo (primero en entrar primero en salir) 2 promedio ponderado	int2	
Btkar_minimo	Cantidad mínima de mercadería que debe existir en bodega	numeric	
Btkar_maximo	Cantidad máxima de mercadería que debe existir en bodega	numeric	

CONTINÚA

Tabla 6.15 (cont.). Descripción de la tabla BSX_TARJETA_KARDEX

Btkar_existencia	Cantidad de artículos existentes en bodega	numeric
Btkar_activo	Indica si el estado de la kardex True - activo False - inactivo	bool
Btkar_fecha_crea	Fecha en la cual se creó la kardex	timestamp
Btkar_observacion	Observación sobre la kardex	varchar(50)
Btkar_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool
Btkar_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)
Btkar_fecha_insert	Fecha y hora en que el usuario inserto los datos.	timestamp with time zone
Btkar_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)
Btkar_fecha_update	Fecha y hora en que el usuario actualizo los datos.	timestamp with time zone

Tabla 6.16. Descripción de la tabla BSX_TKDETALLE.

Tabla: BSX_TKDETALLE			
Nombre	Comentario	Tipo de dato	Clave
Btkde_codigo	Clave primaria de la tabla bsx_tkdetalle	varchar(20)	pk
Btkar_codigo	Clave primaria de la tabla bsx_tarjeta_kardex	varchar(20)	fk
Btkde_tipo_mov	Tipo de movimiento: 0 - entrada 1 - salidas 2 - inventario inicial	int2	
Btkde_tipo_transac	Tipo de transacción 0-compra 1-devolucion venta 2-venta 3-devolucion compra 4-inventario inicial	int2	

CONTINÚA

Tabla 6.16 (cont). Descripción de la tabla BSX_TKDETALLE.

Btkde_fecha	Fecha del movimiento	timestamp with time zone
Btkde_referencia	Código de referencia del movimiento	varchar(50)
Btkde_prov_cliente	Código del proveedor o cliente	varchar(20)
Btkde_concepto	Concepto del movimiento que se registra	varchar(500)
Btkde_cant_entrada	Cantidad de artículos que ingresa a bodega	numeric
Btkde_vu_entrada	Valor unitario del artículo que ingresa a bodega	numeric
Btkde_vtotal_entrada	Valor total de la entrada de artículos	numeric
Btkde_cant_salida	Cantidad de artículos que sale de bodega	numeric
Btkde_vu_salida	Valor unitario del artículo que sale de bodega	numeric
Btkde_vtotal_salida	Valor total de la salida de artículos	numeric
Btkde_existencia	Diferencia entre la cantidad de entrada y la cantidad de salida	numeric
Btkde_valor_u_exist	Valor unitario de la existencia	numeric
Btkde_valor_t_exist	Valor total de las existencias	numeric
Btkde_observacion	Observación sobre el movimiento	text
Btkde_estado_cierre	Estado del inventario True - abierto False - cerrado	bool
Btkde_fecha_cierre	Fecha de cierre de inventario	timestamp with time zone
Btkde_btkde_codigo	Código de la venta o compra devuelta	varchar(20)
Btkde_estado	Estado del registro. True - el registro está activo False - el registro está inactivo, borrado lógico	bool

CONTINÚA

Tabla 6.16 (cont.). Descripción de la tabla BSX_TKDETALLE

Btkde_usuario_insert	Usuario que inserto los datos en la tabla	varchar(50)
Btkde_fecha_insert	Fecha y hora en que el usuario inserto los datos	timestamp with time zone
Btkde_usuario_update	Usuario que actualizo los datos de la tabla	varchar(50)
Btkde_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone

Tabla 6.17. Descripción de la tabla BSX_SERIALES

Tabla: BSX_SERIALES			
Nombre	Comentario	Tipo de dato	Clave
Bseri_codigo	Clave primaria de la tabla bsx_seriales	varchar(20)	pk
Btkar_codigo	Clave primaria de la tabla bsx_tarjeta_kardex	varchar(20)	fk
Btkde_codigo	Clave primaria de la tabla bsx_tkdetalle	varchar(20)	fk
Bsx_btkde_codigo	Codigo del movimiento realizado	varchar(20)	fk
Bseri_serie	Serie del producto	varchar(200)	
Bseri_observacion	Observación sobre la serie	varchar(100)	
Bseri_disponible	Indica si la serie del producto está disponible	bool	
Bseri_estado	Estado del registro.	bool	
Bseri_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Bseri_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Bseri_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Bseri_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.18. Descripción de la tabla BSX_TIPO_FORMA_PAGO.

Tabla: BSX_TIPO_FORMA_PAGO			
Nombre	Comentario	Tipo de dato	Clave
Btfpa_codigo	Clave primaria de la tabla bsx_tipo_forma_pago	varchar(20)	pk
Bplcd_codigo	Clave primaria de la tabla bsx_plan_cuentas_detalle	varchar(20)	Fk
Bagen_codigo	Clave primaria de la tabla bsx_agencia	varchar(20)	Fk
Btfpa_nombre	Nombre de la forma de pago	varchar(50)	
Btfpa_descripcion	Descripcion de la forma de pago	varchar(500)	
Btfpa_estado_tipo	Forma de pago es valida o no	bool	
Btfpa_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool	
Btfpa_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Btfpa_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Btfpa_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Btfpa_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.19. Descripción de la tabla BSX_VENTA.

Tabla: BSX_VENTA			
Nombre	Comentario	Tipo de dato	Clave
Bvent_codigo	Clave primaria de la tabla bsx_venta	varchar(20)	pk
Bclie_codigo	Clave primaria de la tabla bsx_cliente	varchar(20)	fk
Bagen_codigo	Clave primaria de la tabla bsx_agencia	varchar(20)	fk
Bempl_codigo	Clave primaria de la tabla bsx_empleado	varchar(20)	fk

CONTINÚA

Tabla 6.19 (cont.). Descripción de la tabla BSX_VENTA

Bdcom_codigo	Clave primaria de la tabla bsx_documento_comercial	varchar(20)	fk
Bvent_compr_num	Numero del comprobante de venta	varchar(30)	
Bvent_tipo_compr	Tipo de comprobante 0 - factura	int2	
Bvent_estado_venta	Estado de la venta 0 - anulada 1 - vigente	int2	
Bvent_estado_pago	Estado del pago 0 pendiente de pago 1 pagado	int2	
Bvent_tipo_pago	Forma de pago 0 - contado 1 - credito con la empresa	int2	
Bvent_fecha_emision	Fecha y hora en que se realizo la venta	timestamp	
Bvent_porc_iva_uno	Porcentaje del impuesto iva	numeric	
Bvent_valor_iva_uno	Valor del iva	numeric	
Bvent_porc_iva_dos	Porcentaje del impuesto iva	numeric	
Bvent_valor_iva_dos	Valor del iva	numeric	
Bvent_porc_desc	Porcentaje de descuento	numeric	
Bvent_valor_desc	Valor del descuento	numeric	
Bvent_subtotal	Subtotal de la venta	numeric	
Bvent_total	Valor total de la venta	numeric	
Bvent_termino_venta	Termino de la venta	text	
Bvent_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool	
Bvent_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Bvent_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Bvent_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Bvent_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.20. Descripción de la tabla BSX_VENTA_DETALLE.

Tabla: BSX_VENTA_DETALLE			
Nombre	Comentario	Tipo de dato	Clave
Bvdet_codigo	Clave primaria de la tabla bsx_venta_detalle	varchar(20)	pk
Bvent_codigo	Clave primaria de la tabla bsx_venta	varchar(20)	fk
Barti_codigo	Clave primaria de la tabla bsx_articulo	varchar(20)	fk
Bseri_codigo	Clave primaria de la tabla bsx_seriales	varchar(20)	fk
Btkar_codigo	Clave primaria de la tabla bsx_tarjeta_kardex	varchar(20)	fk
Bvdet_cantidad	Cantidad de articulos	numeric	
Bvdet_descripcion	Descripcion del articulo	text	
Bvdet_valor_unit	Valor unitario del articulo	numeric	
Bvdet_valor_desc	Porcentaje de descuento	numeric	
Bvdet_total	Valor total de articulo	numeric	
Bvdet_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool	
Bvdet_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Bvdet_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Bvdet_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Bvdet_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.21. Descripción de la tabla BSX_FORMA_PAGO_VENTA.

Tabla: BSX_FORMA_PAGO_VENTA			
Nombre	Comentario	Tipo de dato	Clave
Bfpve_codigo	Clave primaria de la tabla bsx_forma_pago_venta	varchar(20)	pk
Btfpa_codigo	Clave primaria de la tabla bsx_tipo_forma_pago	varchar(20)	fk
Bvent_codigo	Clave primaria de la tabla bsx_venta	varchar(20)	fk
Bfpve_valor	Valor que se cancela con la formas de pago	numeric	
Bfpve_numero	Numero de cheque emitido o transferencia realizada	varchar(50)	
Bfpve_banco	Banco al que pertenece el cheque o transferencia	varchar(50)	
Bfpve_observacion	Observacion sobre la transaccion realizada	varchar(500)	
Bfpve_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool	
Bfpve_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Bfpve_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Bfpve_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Bfpve_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.22. Descripción de la tabla BSX_RETENCION_VENTA.

Tabla: BSX_RETENCION_VENTA			
Nombre	Comentario	Tipo de dato	Clave
Brete_codigo	Clave primaria de la tabla bsx_retencion_venta	varchar(20)	pk

CONTINÚA

Tabla 6.22 (cont.). Descripción de la tabla BSX_RETENCION_VENTA

Bvent_codigo	Clave primaria de la tabla bsx_venta	varchar(20) fk
Brete_numero	Numero del comprobante de retencion	varchar(50)
Brete_fecha_reten	Fecha en la cual se realiza la retencion	timestamp
Brete_total_reten	Cantidad total de la retención	numeric
Brete_adjunto	Adjunto de la retención	bytea
Brete_nombre_adjunto	Nombre del adjunto	varchar(200)
Brete_tipo_adjunto	Extensión del adjunto	varchar(50)
Brete_observacion	Observación sobre la retención	varchar(200)
Brete_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool
Brete_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)
Brete_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone
Brete_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)
Brete_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone

Tabla 6.23. Descripción de la tabla BSX_RETENCION_VENTA_DETALLE.

Tabla: BSX_RETENCION_VENTA_DETALLE			
Nombre	Comentario	Tipo de dato	Clave
Brvde_codigo	Clave primaria de la tabla bsx_retencion_venta_detalle	varchar(20)	pk

CONTINÚA

Tabla 6.23 (cont.). Descripción de la tabla BSX_RETENCION_VENTA_DETALLE

Brete_codigo	Clave primaria de la tabla bsx_retencion_venta	varchar(20)	fk
Brvde_concepto	Concepto sobre el cual se hace la retención.	varchar(50)	
Brvde_ejer_fiscal	ejercicio fiscal de la retención	varchar(20)	
Brvde_base_imp	Base imponible para la retención	numeric	
Brvde_impuesto	Impuesto que se retiene	varchar(20)	
Brvde_porcentaje	Porcentaje de la retención	numeric	
Brvde_valor	Valor de la retención	numeric	
Brvde_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo	bool	
Brvde_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Brvde_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Brvde_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Brvde_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.24. Descripción de la tabla BSX_PROFORMA.

Tabla: BSX_PROFORMA			
Nombre	Comentario	Tipo de dato	Clave
Bprof_codigo	Clave primaria de la tabla bsx_proforma	varchar(20)	pk
Bparr_codigo	Clave primaria de la tabla bsx_parroquia	varchar(20)	fk
Bagen_codigo	Clave primaria de la tabla bsx_agencia	varchar(20)	fk
Bclie_codigo	Clave primaria de la tabla bsx_cliente	varchar(20)	fk

CONTINÚA

Tabla 6.24 (cont.). Descripción de la tabla BSX_PROFORMA

Bempl_codigo	Clave primaria de la tabla bsx_employed	varchar(20) fk
Bprof_numero	Numero secuencial de la proforma	varchar(20)
Bprof_fecha_emision	Fecha en que se emitió la proforma	timestamp
Bprof_fecha_vencim	Fecha de vencimiento de la proforma	timestamp
Bprof_ci_ruc	Cedula de identidad o ruc del solicitante de la proforma	varchar(13)
Bprof_nombre	Nombre del solicitante de la proforma	varchar(50)
Bprof_apellido	Apellidos del solicitante de la proforma	varchar(50)
Bprof_direccion	Dirección del solicitante de la proforma	varchar(500)
Bprof_telefono	Teléfono del solicitante de la proforma	varchar(100)
Bprof_subtotal	Valor subtototal de la proforma	numeric
Bprof_porcent_iva	Porcentaje del iva	numeric
Bprof_iva	Valor del iva	numeric
Bprof_total	Valor total de la proforma	numeric
Bprof_observacion	Observación de la proforma	text
Bprof_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool
Bprof_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)
Bprof_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone
Bprof_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)
Bprof_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone

Tabla 6.25. Descripción de la tabla BSX_PROFORMA_DETALLE.

Tabla: BSX_PROFORMA_DETALLE			
Nombre	Comentario	Tipo de dato	Clave
Bpdet_codigo	Clave primaria de la tabla bsx_proforma_detalle	varchar(20)	pk
Bprof_codigo	Clave primaria de la tabla bsx_proforma	varchar(20)	fk
Barti_codigo	Clave primaria de la tabla bsx_articulo	varchar(20)	fk
Bpdet_cantidad	Cantidad de artículos	numeric	
Bpdet_valor_unit	Valor unitario del articulo	numeric	
Bpdet_valor_desc	Porcentaje de descuento	numeric	
Bpdet_total	Valor total de los artículos	numeric	
Bpdet_observacion	Observación sobre los artículos	varchar(100)	
Bprof_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo, borrado lógico	bool	
Bpdet_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)	
Bpdet_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone	
Bpdet_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)	
Bpdet_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone	

Tabla 6.26. Descripción de la tabla BSX_NOTA.

Tabla: BSX_NOTA			
Nombre	Comentario	Tipo de dato	Clave
Bnota_codigo	Clave primaria de la tabla bsx_nota	varchar(20)	pk
Bclie_codigo	Clave primaria de la tabla bsx_cliente	varchar(20)	fk
Bvent_codigo	Clave primaria de la tabla bsx_venta	varchar(20)	fk
Bdcom_codigo	Clave primaria de la tabla bsx_documento_comercial	varchar(20)	fk
Bagen_codigo	Clave primaria de la tabla bsx_agencia	varchar(20)	fk
Bnota_tipo	Tipo del documento: 0 nota de credito 1 nota de debito	int2	
Bnota_num_comprob	numero de comprobante	varchar(20)	
Bnota_fecha_emision	Fecha en la que se emite el documento	timestamp	
Bnota_fecha_caduca	Fecha en la que caduca el documento	date	
Bnota_subtotal	Valor subtotal	numeric	
Bnota_porct_iva	Porcentaje del iva	numeric	
Bnota_iva	Valor del iva	numeric	
Bnota_porct_desc	Porcentaje de descuento	numeric	
Bnota_valor_desc	Valor de descuento	numeric	
Bnota_valor_total	Valor total	numeric	
Bnota_estado_nota	Estado de la nota: 0 - no aplicada 1 - aplicado 2 - anulado	int2	
Bnota_adjunto	Adjunto del documento	bytea	
Bnota_tipo_adjunto	Extensión de adjunto	varchar(20)	
Bnota_nombre_adjunto	Nombre del adjunto	varchar(50)	

CONTINÚA

Tabla 6.26 (cont.). Descripción de la tabla BSX_NOTA.

Bnota_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo	bool
Bnota_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)
Bnota_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone
Bnota_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)
Bnota_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone

Tabla 6.27. Descripción de la tabla BSX_DETALLE_NOTA.

Tabla: BSX_DETALLE_NOTA			
Nombre	Comentario	Tipo de dato	Clave
Bdeno_codigo	Clave primaria de la tabla bsx_detalle_notas	varchar(20)	pk
Bnota_codigo	Clave primaria de la tabla bsx_notas	varchar(20)	fk
Bmdve_codigo	Clave primaria de la tabla bsx_motivo_devol_venta	varchar(20)	fk
Bvdet_codigo	Clave primaria de la tabla bsx_venta_detalle	varchar(20)	fk
Bdeno_cantidad	Cantidad de articulos devueltos	numeric	
Bdeno_total	Valor total del articulo devuelto	numeric	
Bdeno_estado	Estado del registro. True - el registro esta activo False - el registro está inactivo	bool	

CONTINÚA

Tabla 6.27 (cont.). Descripción de la tabla BSX_DETALLE_NOTA.

Bdeno_usuario_insert	Usuario que inserto datos en la tabla	varchar(50)
Bdeno_fecha_insert	Fecha y hora en que el usuario inserto datos	timestamp with time zone
Bdeno_usuario_update	Usuario que actualizo datos de la tabla	varchar(50)
Bdeno_fecha_update	Fecha y hora en que el usuario actualizo los datos	timestamp with time zone

6.8.4. Implementación

En base a la arquitectura de software SOA se desarrolló el módulo de gestión de inventarios y facturación utilizando PostgreSQL como gestor de base de datos, java como lenguaje de programación en el nivel de servicios y en el nivel de clientes se utilizó extjs, javaScript y ajax.

A continuación se presenta la codificación de un proceso del módulo de gestión de inventarios y facturación que sirve como referencia para los demás ya desarrollados.

Proceso de guardado de datos de la tabla BSX_RETENCION_VENTA

Funciones sql

Función para crear la clave primaria de la tabla

```

/**
 * FUNCION QUE GENERA CLAVES PARA LAS TABLAS DEL ESQUEMA.
 * @param INTABLENAME NOMBRE DE LA TABLA QUE REQUIERE LA
 * CLAVE.
 * @param INCOLNAME COLUMNA DE LA TABLA QUE REQUIERE LA
 * CLVAE.
 * @param INABR ABREVIATURA DEL NOMBRE DE LA TABLA
 * SOLICITANTE.
 * @return NUEVA CLAVE GENERADA.
 */
CREATE OR REPLACE FUNCTION SCH_BSX.F_KEYGEN(

```

```

IN INTABLENAME VARCHAR,
IN INCOLNAME VARCHAR,
IN INABR VARCHAR
)RETURNS VARCHAR AS
$$
DECLARE
TMP_NUMREGISTROS NUMERIC;
TMP_EXISTS NUMERIC;
OUT_KEY VARCHAR;
BEGIN
EXECUTE 'SELECT COUNT('||INCOLNAME||')+1 FROM
SCH_BSX.'||INTABLENAME INTO TMP_NUMREGISTROS;
OUT_KEY := INABR||TMP_NUMREGISTROS;
EXECUTE 'SELECT COUNT('||INCOLNAME||') FROM
SCH_BSX.'||INTABLENAME||' WHERE '||INCOLNAME||' = $1' INTO
TMP_EXISTS USING OUT_KEY;
IF (TMP_EXISTS > 0) THEN
FOR I IN 1..TMP_NUMREGISTROS LOOP
EXECUTE 'SELECT COUNT('||INCOLNAME||') FROM
SCH_BSX.'||INTABLENAME||' WHERE '||INCOLNAME||' = $1' INTO
TMP_EXISTS USING INABR||I;
IF (TMP_EXISTS = 0) THEN
OUT_KEY := INABR||I;
EXIT;
END IF;
END LOOP;
END IF;
RETURN OUT_KEY;
END
$$
LANGUAGE 'PLPGSQL' VOLATILE CALLED ON NULL INPUT;

```

Función para la inserción de datos

```

/**
* FUNCION SCH_BSX.F_BSX_RETENCION_VENTA_INSERT
* INGRESA UN REGISTRO EN LA TABLA
SCH_BSX.BSX_RETENCION_VENTA
* @param IN_BRETE_CODIGO CLAVE PRIMARIA DE LA TABLA
BSX_RETENCION_VENTA
* @param IN_BVENT_CODIGO CLAVE PRIMARIA DE LA TABLA
BSX_VENTA
* @param IN_BRETE_NUMERO NUMERO DEL COMPROBANTE DE RETENCION
* @param IN_BRETE_FECHA_RETEN FECHA EN LA CUAL SE REALIZA LA
RETENCION
* @param IN_BRETE_TOTAL_RETEN VALOR TOTAL DE LA RETENCION
* @param IN_BRETE_ADJUNTO DOCUMENTO ADJUNTO DE A LA
RETENCION
* @param IN_BRETE_NOMBRE_ADJUNTO NOMBRE DEL DOCUMENTO
ADJUNTO
* @param IN_BRETE_TIPO_ADJUNTO EXTENCION DEL DOCUMENTO
ADJUNTO

```

```

* @param IN_BRETE_ESTADO ESTADO DEL REGISTRO.
  TRUE CUANDO EL REGISTRO ESTA ACTIVO
  FALSE CUANDO EL REGISTRO ESTA INACTIVO, BORRADO LOGICO
* @param IN_BRETE_USUARIO_INSERT USUARIO QUE INSERTO DATOS
SOBRE LA TABLA
* @param IN_BRETE_USUARIO_UPDATE USUARIO QUE ACTUALIZO DATOS
SOBRE LA TABLA
* @return VARCHAR
**/
CREATE OR REPLACE FUNCTION
SCH_BSX.F_BSX_RETENCION_VENTA_INSERT(
  IN IN_BRETE_CODIGO
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_CODIGO%TYPE,
  IN IN_BVENT_CODIGO
  SCH_BSX.BSX_RETENCION_VENTA.BVENT_CODIGO%TYPE,
  IN IN_BRETE_NUMERO
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_NUMERO%TYPE,
  IN IN_BRETE_FECHA_RETEN
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_FECHA_RETEN%TYPE,
  IN IN_BRETE_TOTAL_RETEN
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_TOTAL_RETEN%TYPE,
  IN IN_BRETE_ADJUNTO
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_ADJUNTO%TYPE,
  IN IN_BRETE_NOMBRE_ADJUNTO
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_NOMBRE_ADJUNTO%TYPE,
  IN IN_BRETE_TIPO_ADJUNTO
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_TIPO_ADJUNTO%TYPE,
  IN IN_BRETE_OBSERVACION
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_OBSERVACION%TYPE,
  IN IN_BRETE_ESTADO
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_ESTADO%TYPE,
  IN IN_BRETE_USUARIO_INSERT
  SCH_BSX.BSX_RETENCION_VENTA.BRETE_USUARIO_INSERT%TYPE
) RETURNS VARCHAR AS
$$
DECLARE
OUT_KEY VARCHAR;
BEGIN
  IF (UPPER(IN_BRETE_CODIGO) = 'KEYGEN') THEN
    SELECT
SCH_BSX.F_KEYGEN('BSX_RETENCION_VENTA', 'BRETE_CODIGO', 'BRETE
') INTO OUT_KEY;
  ELSE
    OUT_KEY := IN_BRETE_CODIGO;
  END IF;
  INSERT INTO SCH_BSX.BSX_RETENCION_VENTA(
    BRETE_CODIGO,
    BVENT_CODIGO,
    BRETE_NUMERO,
    BRETE_FECHA_RETEN,
    BRETE_TOTAL_RETEN,
    BRETE_ADJUNTO,
    BRETE_NOMBRE_ADJUNTO,

```

```

        BRETE_TIPO_ADJUNTO,
        BRETE_OBSERVACION,
        BRETE_ESTADO,
        BRETE_USUARIO_INSERT,
        BRETE_FECHA_INSERT,
        BRETE_USUARIO_UPDATE,
        BRETE_FECHA_UPDATE
    ) VALUES (
        OUT_KEY,
        IN_BVENT_CODIGO,
        IN_BRETE_NUMERO,
        IN_BRETE_FECHA_RETEN,
        IN_BRETE_TOTAL_RETEN,
        IN_BRETE_ADJUNTO,
        IN_BRETE_NOMBRE_ADJUNTO,
        IN_BRETE_TIPO_ADJUNTO,
        IN_BRETE_OBSERVACION,
        IN_BRETE_ESTADO,
        IN_BRETE_USUARIO_INSERT,
        CURRENT_TIMESTAMP,
        IN_BRETE_USUARIO_INSERT,
        CURRENT_TIMESTAMP
    );
    RETURN OUT_KEY;
END
$$
LANGUAGE 'plpgsql' VOLATILE CALLED ON NULL INPUT;

```

Clases java

Clase de objetos

```

package com.besixplus.sisac.objects;
import java.io.Serializable;

/**
 * CLASE: BSX_RETENCION_VENTA
 * TABLA: BSX_RETENCION_VENTA
 * DESCRIPCION: ALMACENA LOS DATOS DE LA RETENCION
 * ABREVIATURA: BRETE
 */

public class Bsx_retencion_venta implements Serializable{
    private static final long serialVersionUID = 2090396319;

    /**
     * CLAVE PRIMARIA DE LA TABLA BSX_RETENCION_VENTA
     */
    private String myBrete_codigo;

    /**

```

```

* CLAVE PRIMARIA DE LA TABLA BSX_VENTA
*/
private String myBvent_codigo;

/**
* NUMERO DEL COMPROBANTE DE RETENCION
*/
private String myBrete_numero;

/**
* FECHA EN LA CUAL SE REALIZA LA RETENCION
*/
private java.util.Date myBrete_fecha_reten;

/**
* CANTIDAD TOTAL DE LA RETENCION
*/
private java.math.BigDecimal myBrete_total_reten;

/**
* DOCUMENTO ADJUNTO DE A LA RETENCION
*/
private byte[] myBrete_adjunto;

/**
* NOMBRE DEL DOCUMENTO ADJUNTO
*/
private String myBrete_nombre_adjunto;

/**
* EXTENSION DEL DOCUMENTO ADJUNTO
*/
private String myBrete_tipo_adjunto;

/**
* OBSERVACION SOBRE LA RETENCION
*/
private String myBrete_observacion;

/**
* ESTADO DEL REGISTRO.
* TRUE CUANDO EL REGISTRO ESTA ACTIVO
* FALSE CUANDO EL REGISTRO ESTA INACTIVO, BORRADO LOGICO
*/
private boolean myBrete_estado;

/**
* USUARIO QUE INSERTO DATOS SOBRE LA TABLA
*/
private String myBrete_usuario_insert;

/**
* USUARIO QUE ACTUALIZO DATOS SOBRE LA TABLA

```

```

*/
private String myBrete_usuario_update;

/**
 * CONSTRUCTOR DE LA CLASE Bsx_venta
 */
public Bsx_retencion_venta() {}

/**
 * CONSTRUCTOR DE LA CLASE Bsx_venta
 * @param inBrete_codigo CLAVE PRIMARIA DE LA TABLA
BSX_RETENCION_VENTA
 * @param inBvent_codigo CLAVE PRIMARIA DE LA TABLA
BSX_VENTA
 * @param inBrete_numero NUMERO DEL COMPROBANTE DE
RETENCION
 * @param inBrete_fecha_reten FECHA EN LA CUAL SE REALIZA
LA RETENCION
 * @param inBrete_total_reten VALOR TOTAL DE LA RETENCION
 * @param inBrete_adjunto DOCUMENTO ADJUNTO DE A LA
RETENCION
 * @param inBrete_nombre_adjunto NOMBRE DEL DOCUMENTO
ADJUNTO
 * @param inBrete_tipo_adjunto EXTENCION DEL DOCUMENTO
ADJUNTO
 * @param inBrete_observacion OBSERVACION SOBRE LA
RETENCION
 * @param inBvent_estado ESTADO DEL REGISTRO.
TRUE CUANDO EL REGISTRO ESTA ACTIVO
FALSE CUANDO EL REGISTRO ESTA INACTIVO, BORRADO LOGICO
 * @param inBvent_usuario_insert USUARIO QUE INSERTO DATOS
SOBRE LA TABLA
 * @param inBvent_usuario_update USUARIO QUE ACTUALIZO
DATOS SOBRE LA TABLA
 */

public Bsx_retencion_venta(
    String inBrete_codigo,
    String inBvent_codigo,
    String inBrete_numero,
    java.util.Date inBrete_fecha_reten,
    java.math.BigDecimal inBrete_total_reten,
    byte[] inBrete_adjunto,
    String inBrete_nombre_adjunto,
    String inBrete_tipo_adjunto,
    String inBrete_observacion,
    boolean inBrete_estado,
    String inBrete_usuario_insert,
    String inBrete_usuario_update
) {
    this.setBRETE_CODIGO(inBrete_codigo);
    this.setBVENT_CODIGO(inBvent_codigo);
    this.setBRETE_NUMERO(inBrete_numero);

```

```

    this.setBRETE_FECHA_RETEN(inBrete_fecha_reten);
    this.setBRETE_TOTAL_RETEN(inBrete_total_reten);
    this.setBRETE_ADJUNTO(inBrete_adjunto);
    this.setBRETE_NOMBRE_ADJUNTO(inBrete_nombre_adjunto);
    this.setBRETE_TIPO_ADJUNTO(inBrete_tipo_adjunto);
    this.setBRETE_OBSERVACION(inBrete_observacion);
    this.setBRETE_ESTADO(inBrete_estado);
    this.setBRETE_USUARIO_INSERT(inBrete_usuario_insert);
    this.setBRETE_USUARIO_UPDATE(inBrete_usuario_update);
}

/**
 * ESTABLECE LA CLAVE PRIMARIA DE LA TABLA
 * BSX_RETENCION_VENTA
 * @param inBrete_codigo CLAVE PRIMARIA DE LA TABLA
 * BSX_RETENCION_VENTA
 */
public void setBRETE_CODIGO(String inBrete_codigo){
    this.myBrete_codigo = inBrete_codigo;
}

/**
 * OBTIENE LA CLAVE PRIMARIA DE LA TABLA
 * BSX_RETENCION_VENTA
 * @return String CLAVE PRIMARIA DE LA TABLA
 * BSX_RETENCION_VENTA */
public String getBRETE_CODIGO(){
    return this.myBrete_codigo;
}

/**
 * ESTABLECE LA CLAVE PRIMARIA DE LA TABLA BSX_VENTA
 * @param inBvent_codigo CLAVE PRIMARIA DE LA TABLA
 * BSX_VENTA
 */
public void setBVENT_CODIGO(String inBvent_codigo){
    this.myBvent_codigo = inBvent_codigo;
}

/**
 * OBTIENE LA CLAVE PRIMARIA DE LA TABLA BSX_VENTA
 * @return String CLAVE PRIMARIA DE LA TABLA BSX_VENTA
 */
public String getBVENT_CODIGO(){
    return this.myBvent_codigo;
}

/**
 * ESTABLECE EL NUMERO DEL COMPROBANTE DE RETENCION
 * @param inBrete_numero NUMERO DEL COMPROBANTE DE
 * RETENCION
 */
public void setBRETE_NUMERO(String inBrete_numero){
    this.myBrete_numero = inBrete_numero;
}

```



```

}
/**
 * OBTIENE EL NUMERO DEL COMPROBANTE DE RETENCION
 * @return String NUMERO DEL COMPROBANTE DE RETENCION
 */
public String getBRETE_NUMERO(){
    return this.myBrete_numero;
}

/**
 * ESTABLECE LA FECHA EN LA CUAL SE REALIZA LA RETENCION
 * @param inBrete_fecha_reten FECHA EN LA CUAL SE REALIZA
LA RETENCION
 */
public void setBRETE_FECHA_RETEN(java.util.Date
inBrete_fecha_reten){
    this.myBrete_fecha_reten = inBrete_fecha_reten;
}

/**
 * OBTIENE LA FECHA EN LA CUAL SE REALIZA LA RETENCION
 * @return java.util.Date FECHA EN LA CUAL SE REALIZA LA
RETENCION
 */
public java.util.Date getBRETE_FECHA_RETEN(){
    return this.myBrete_fecha_reten;
}

/**
 * ESTABLECE EL VALOR TOTAL DE LA RETENCION
 * @param inBrete_total_reten VALOR TOTAL DE LA RETENCION
 */
public void setBRETE_TOTAL_RETEN(java.math.BigDecimal
inBrete_total_reten){
    this.myBrete_total_reten = inBrete_total_reten;
}

/**
 * OBTIENE EL VALOR TOTAL DE LA RETENCION
 * @return java.math.BigDecimal VALOR TOTAL DE LA RETENCION
 */
public java.math.BigDecimal getBRETE_TOTAL_RETEN(){
    return this.myBrete_total_reten;
}

/**
 * ESTABLECE EL DOCUMENTO ADJUNTO DE A LA RETENCION
 * @param inBrete_adjunto DOCUMENTO ADJUNTO DE A LA
RETENCION
 * */
public void setBRETE_ADJUNTO(byte[] inBrete_adjunto){
    this.myBrete_adjunto = inBrete_adjunto ;
}

```

```

}

/**
 * OBTIENE EL DOCUMENTO ADJUNTO DE A LA RETENCION
 * return java.math.BigDecimal DOCUMENTO ADJUNTO DE A LA
 * RETENCION
 * */
public byte[] getBRETE_ADJUNTO(){
    return this.myBrete_adjunto;
}

/**
 * ESTABLECE EL NOMBRE DEL DOCUMENTO ADJUNTO
 * @param inBrete_nombre_adjunto NOMBRE DEL DOCUMENTO
 * ADJUNTO
 * */
public void setBRETE_NOMBRE_ADJUNTO(String
inBrete_nombre_adjunto){
    this.myBrete_nombre_adjunto = inBrete_nombre_adjunto;
}

/**
 * OBTIENE EL NOMBRE DEL DOCUMENTO ADJUNTO
 * @return String NOMBRE DEL DOCUMENTO ADJUNTO
 * */
public String getBRETE_NOMBRE_ADJUNTO(){
    return this.myBrete_nombre_adjunto;
}

/**
 * ESTABLECE LA EXTENCION DEL ADJUNTO
 * @param inBrete_nombre_adjunto EXTENCION DEL ADJUNTO
 * */
public void setBRETE_TIPO_ADJUNTO(String
inBrete_tipo_adjunto){
    this.myBrete_tipo_adjunto = inBrete_tipo_adjunto;
}

/**
 * OBTIENE EXTENCION DEL ADJUNTO
 * @return String NOMBRE DEL ADJUNTO
 * */
public String getBRETE_TIPO_ADJUNTO(){
    return this.myBrete_tipo_adjunto;
}

/**
 * ESTABLECE OBSERVACION SOBRE LA RETENCION
 * @param inBrete_observacion OBSERVACION SOBRE LA
 * RETENCION
 * */
public void setBRETE_OBSERVACION(String
inBrete_observacion){
    this.myBrete_observacion = inBrete_observacion;
}

```

```

}

/**
 * OBTIENE OBSERVACION SOBRE LA RETENCION
 * @return String OBSERVACION SOBRE LA RETENCION
 */
public String getBRETE_OBSERVACION(){
    return this.myBrete_observacion;
}

/**
 * ESTABLECE ESTADO DEL REGISTRO.
 TRUE CUANDO EL REGISTRO ESTA ACTIVO
 FALSE CUANDO EL REGISTRO ESTA INACTIVO, BORRADO LOGICO
 * @param inbvent_estado ESTADO DEL REGISTRO
 */
public void setBRETE_ESTADO(boolean inBrete_estado){
    this.myBrete_estado = inBrete_estado;
}

/**
 * OBTIENE ESTADO DEL REGISTRO.
 TRUE CUANDO EL REGISTRO ESTA ACTIVO
 FALSE CUANDO EL REGISTRO ESTA INACTIVO, BORRADO LOGICO
 @return boolean ESTADO DEL REGISTRO.
 */
public boolean getBRETE_ESTADO(){
    return this.myBrete_estado;
}

/**
 * ESTABLECE USUARIO QUE INSERTO DATOS SOBRE LA TABLA
 * @param inbvent_usuario_insert USUARIO QUE INSERTO DATOS
 SOBRE LA TABLA
 */
public void setBRETE_USUARIO_INSERT(String
inBrete_usuario_insert){
    this.myBrete_usuario_insert = inBrete_usuario_insert;
}

/**
 * OBTIENE USUARIO QUE INSERTO DATOS SOBRE LA TABLA
 * @return String USUARIO QUE INSERTO DATOS SOBRE LA TABLA
 */
public String getBRETE_USUARIO_INSERT(){
    return this.myBrete_usuario_insert;
}

/**
 * ESTABLECE USUARIO QUE ACTUALIZO DATOS SOBRE LA TABLA
 * @param inbvent_usuario_update USUARIO QUE ACTUALIZO
 DATOS SOBRE LA TABLA
 */

```

```

public void setBRETE_USUARIO_UPDATE(String
inBrete_usuario_update){
    this.myBrete_usuario_update = inBrete_usuario_update;
}

/**
 * OBTIENE USUARIO QUE ACTUALIZO DATOS SOBRE LA TABLA
 * @return String USUARIO QUE ACTUALIZO DATOS SOBRE LA
TABLA
 */
public String getBRETE_USUARIO_UPDATE(){
    return this.myBrete_usuario_update;
}
}

```

Clase db

```

package com.besixplus.sisac.db;
import java.util.ArrayList;
import java.util.HashMap;
import java.io.Serializable;
import java.sql.CallableStatement;
import java.sql.ResultSet;
import java.sql.Savepoint;
import java.sql.SQLException;
import java.sql.Types;

/**
 * CLASE: BSX_RETENCION_VENTA
 * TABLA: BSX_RETENCION_VENTA
 * DESCRIPCION: ALMACENA LOS DATOS DE LA RETENCION
 * ABREVIATURA: BRETE
 */
public class Bsx_retencion_venta implements Serializable{
    private static final long serialVersionUID = 2082434238;

    /**
     * VARIABLE QUE ALMACENA UNA INSTANCIA DE LA CLASE
     com.besixplus.sisac.objects.Bsx_retencion_venta.
     */
    private com.besixplus.sisac.objects.Bsx_retencion_venta
myBsx_retencion_venta = null;

    /**
     * CONSTRUCTOR DE LA CLASE Bsx_retencion_venta.
     */
    public Bsx_retencion_venta(){

    /**
     * CONSTRUCTOR DE LA CLASE Bsx_retencion_venta.
     * @param inBsx_retencion_venta OBJETO QUE CONTIENE TODA LA

```

```

INFORMACION NECESARIA PARA INSERTAR EN LA TABLA
Bsx_retencion_venta.
*/
public Bsx_retencion_venta(
    com.besixplus.sisac.objects.Bsx_retencion_venta
    inBsx_retencion_venta
){
    this.setBsx_retencion_venta(inBsx_retencion_venta);
}

/**
 * ESTABLECE LA INSTANCIA DEL OBJETO BSX_RETENCION_VENTA.
 * @param inBsx_retencion_venta OBJETO Bsx_retencion_venta.
 * @return void
 */
public void setBsx_retencion_venta(com.besixplus.sisac.
objects.Bsx_retencion_venta inBsx_retencion_venta){
    this.myBsx_retencion_venta = inBsx_retencion_venta;
}

/**
 * OBTIENE LA INSTANCIA DEL OBJETO Bsx_retencion_venta.
 * @return Bsx_retencion_venta OBJETO Bsx_retencion_venta.
 */
public com.besixplus.sisac.objects.Bsx_retencion_venta
getBsx_retencion_venta(){
    return this.myBsx_retencion_venta;
}

/**
 * INSERTA UN REGISTRO EN LA TABLA BSX_RETENCION_VENTA.
 * @param inConnection CONEXION A LA BASE DE DATOS.
 * @return bool <code>>true</code> SI SE INGRESO EL
REGISTRO, CASO CONTRARIO <code>>false</code>.
 */

public String insert(
    java.sql.Connection inConnection
){
    String outResult = "true";
try{
        CallableStatement stmInsert =
        inConnection.prepareCall("{ ? = call
SCH_BSX.F_BSX_RETENCION_VENTA_INSERT(?, ?, ?, ?::date,
?, ?, ?, ?, ?, ?, ?) }");
        stmInsert.registerOutParameter(1,java.sql.Types.VARCHAR)
        ;
        stmInsert.setString(2,this.getBsx_retencion_venta().getB
RETE_CODIGO());
        stmInsert.setString(3,this.getBsx_retencion_venta().getB
VENT_CODIGO());
        stmInsert.setString(4,this.getBsx_retencion_venta().getB
RETE_NUMERO());
    }
}

```

```

stmInsert.setTimestamp(5, this.getBsx_retencion_venta().getBRETE_FECHA_RETEN() != null ? new
java.sql.Timestamp(this.getBsx_retencion_venta().getBRETE_FECHA_RETEN().getTime()) : null);
stmInsert.setBigDecimal(6, this.getBsx_retencion_venta().getBRETE_TOTAL_RETEN());
stmInsert.setBytes(7, this.getBsx_retencion_venta().getBRETE_ADJUNTO());
stmInsert.setString(8, this.getBsx_retencion_venta().getBRETE_NOMBRE_ADJUNTO());
stmInsert.setString(9, this.getBsx_retencion_venta().getBRETE_TIPO_ADJUNTO());
stmInsert.setString(10, this.getBsx_retencion_venta().getBRETE_OBSERVACION());
stmInsert.setBoolean(11, this.getBsx_retencion_venta().getBRETE_ESTADO());
stmInsert.setString(12, this.getBsx_retencion_venta().getBRETE_USUARIO_INSERT());
stmInsert.execute();
this.getBsx_retencion_venta().setBRETE_CODIGO(stmInsert.getString(1));
stmInsert.close();
} catch (SQLException e) {
    com.besixplus.sisac.db.SQLErrorHandler.errorHandler(e);
    outResult = e.getMessage();
}
}
return outResult;
}

```

Clase de servlets

```

package com.besixplus.sisac.servlets;
import java.io.IOException;
import java.io.Serializable;
import java.sql.Connection;
import java.sql.SQLException;
import java.sql.Timestamp;
import java.util.Iterator;
import java.util.List;
import javax.servlet.ServletException;
import javax.servlet.ServletOutputStream;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileUploadException;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;
import java.io.FileNotFoundException;

```

```

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

public class Bsx_retencion_venta extends HttpServlet
implements Serializable{
    private static final long serialVersionUID = 1469076593;

    public Bsx_retencion_venta() {
        super();
    }
    protected void service(HttpServletRequest request,
    HttpServletResponse response) throws ServletException,
    IOException {
    boolean tmpIsMultipart =
    ServletFileUpload.isMultipartContent(request);
    String outResult = "true";
    String tmpRequest = null;
    String tmpDetalleRetencion = null;
    if(tmpIsMultipart){
        com.besixplus.sisac.objects.Bsx_retencion_venta obj =
        new com.besixplus.sisac.objects.Bsx_retencion_venta();
        DiskFileItemFactory factory = new DiskFileItemFactory();
        ServletFileUpload upload = new
        ServletFileUpload(factory);
        upload.setSizeMax(10485760);
        List items;
    try {
        items = upload.parseRequest(request);
        Iterator iter = items.iterator();
        while (iter.hasNext()) {
            FileItem item = (FileItem) iter.next();
            if (item.isFormField()) {
                if(item.getFieldName().equals("request"))
                    tmpRequest = item.getString();
                if(item.getFieldName().equals("brete_codigo"))
                    obj.setBRETE_CODIGO(item.getString());
                if(item.getFieldName().equals("bvent_codigo")
                    && item.getString().trim().length() > 0)
                    obj.setBVENT_CODIGO(item.getString());
                if(item.getFieldName().equals("brete_numero") )
                    obj.setBRETE_NUMERO(item.getString());
                if(item.getFieldName().equals("brete_fecha_reten")
                ){
                    obj.setBRETE_FECHA_RETEN(java.sql.Date.valueOf(it
                    em.getString()));
                }
                if(item.getFieldName().equals("brete_total_reten"))
                    obj.setBRETE_TOTAL_RETEN(java.math.BigDecimal.valueO
                    f(Double.valueOf(item.getString())));
                if(item.getFieldName().equals("brete_observacion") )
                    obj.setBRETE_OBSERVACION(item.getString());
                if(item.getFieldName().equals("brete_nombre_adjunto"

```

```

    ))
    obj.setBRETE_NOMBRE_ADJUNTO(item.getString());
if(item.getFieldName().equals("brete_tipo_adjunto")
)
    obj.setBRETE_TIPO_ADJUNTO(item.getString());

    if(item.getFieldName().equals("retencion_detalle"
) )
        tmpDetalleRetencion = item.getString();

    } else {

if(item.getFieldName().equals("filBrete_adjunto")){
        if(item.get().length > 0){
            obj.setBRETE_ADJUNTO(item.get());
        }
    }
}

obj.setBRETE_ESTADO(true);
Connection tmpCon =
com.besixplus.sisac.db.ManagerConnection.getConnection();
try{
    tmpCon.setAutoCommit(false);
    if(tmpRequest.trim().toUpperCase().equals("INSERT")){
        obj.setBRETE_CODIGO("KEYGEN");
        obj.setBRETE_USUARIO_INSERT(request.getUserPrincipal()
.getName());
        outResult = new
com.besixplus.sisac.db.Bsx_retencion_venta(obj).insert
(tmpCon);
    }else{
        obj.setBRETE_USUARIO_UPDATE(request.getUserPrincipal()
.getName());
        outResult = new
com.besixplus.sisac.db.Bsx_retencion_venta(obj).update
(tmpCon);
    }
    if(outResult.equalsIgnoreCase("true")){
com.besixplus.sisac.objects.Bsx_retencion_venta_deta
lle
objDetalleRetencion = new
com.besixplus.sisac.objects.Bsx_retencion_venta_detalle(
);
objDetalleRetencion.setBRETE_CODIGO(obj.getBRETE_CODIGO(
));boolean tmpRes = new
com.besixplus.sisac.db.Bsx_retencion_venta_detalle(objDe
talleRetencion).deleteBSX_RETENCION_VENTA(tmpCon,false);
        outResult = String.valueOf(tmpRes);
        JSONArray tmpJSONDetalleRetencion = new
JSONArray(tmpDetalleRetencion);
        for (int i= 0 ; i < tmpJSONDetalleRetencion.length()
;i++){

```



```

        JSONObject tmpDetalle = (JSONObject)
tmpJSONDetalleRetencion.get(i);
        objDetalleRetencion = new
com.besixplus.sisac.objects.Bsx_retencion_venta_detalle(
);
        if(tmpDetalle.getString("BRVDE_CODIGO").length() > 0){
            objDetalleRetencion.setBRVDE_CODIGO("KEYGEN");
            objDetalleRetencion.setBRETE_CODIGO(obj.getBRETE_CODIGO(
));

            objDetalleRetencion.setBRVDE_CONCEPTO(tmpDetalle.getSt
tring("BRVDE_CONCEPTO"));
            objDetalleRetencion.setBRVDE_EJER_FISCAL(tmpDetalle.ge
tString("BRVDE_EJER_FISCAL"));
            objDetalleRetencion.setBRVDE_BASE_IMP(java.math.BigDec
imal.valueOf(Double.valueOf(tmpDetalle.getString("BRVDE_
BASE_IMP"))));
            objDetalleRetencion.setBRVDE_IMPUESTO(tmpDetalle.getSt
ring("BRVDE_IMPUESTO").toUpperCase());
            objDetalleRetencion.setBRVDE_PORCENTAJE(java.math.BigD
ecimal.valueOf(Double.valueOf(tmpDetalle.getString("BRVD
E_PORCENTAJE"))));
            objDetalleRetencion.setBRVDE_VALOR(java.math.BigDecima
l.valueOf(Double.valueOf(tmpDetalle.getString("BRVDE_VAL
OR"))));
            objDetalleRetencion.setBRVDE_ESTADO(true);
            objDetalleRetencion.setBRVDE_USUARIO_INSERT(request.getU
serPrincipal().getName());
            objDetalleRetencion.setBRVDE_USUARIO_UPDATE(request.getU
serPrincipal().getName());
            outResult = new
com.besixplus.sisac.db.Bsx_retencion_venta_detalle(objDe
talleRetencion).insert(tmpCon);
        }

if(!outResult.equals("true"))
    break;
    }
}

if(!outResult.equals("true")){
    outResult="false";
}
if(outResult.equalsIgnoreCase("true"))
    tmpCon.commit();
else
    tmpCon.rollback();
    tmpCon.setAutoCommit(true);
try {
    tmpCon.close();
} catch (SQLException e) {
    com.besixplus.sisac.db.SQLErrorHandler.errorHandler(e);
    response.sendError(HttpServletResponse.

```

```

        SC_INTERNAL_SERVER_ERROR, e.getMessage());
        outResult = e.getMessage();
    } catch (SQLException e) {
        com.besixplus.sisac.db.SQLiteErrorHandler.errorHandler(e);
        response.sendError(HttpServletResponse.SC_INTERNAL_SERVER_ERROR, e.getMessage());
        outResult = "false";
    } catch (JSONException e) {
        e.printStackTrace();
    }

    } catch (FileUploadException e) {
        e.printStackTrace();
        response.sendError(HttpServletResponse.SC_INTERNAL_SERVER_ERROR, e.getMessage());
        outResult = "false";
    }
    response.getWriter().write(outResult);
}
else{
    try{
        String tmpCodigo = request.getParameter("codigo");
        String tmpColumna = request.getParameter("columna");
        String tmpReq = request.getParameter("request");
        if(tmpCodigo != null){
            com.besixplus.sisac.objects.Bsx_retencion_venta
            obj = null;
            obj = new com.besixplus.sisac.objects.Bsx_retencion_venta();
            obj.setBRETE_CODIGO(tmpCodigo);
            Connection tmpCon =
            com.besixplus.sisac.db.ManagerConnection.getConnection();
            try{
                tmpCon.setAutoCommit(!com.besixplus.sisac.db.ManagerConnection.isDeployed());
                new com.besixplus.sisac.db.Bsx_retencion_venta
                (obj).select(tmpCon);
                tmpCon.close();
            } catch (SQLException e) {
                com.besixplus.sisac.db.SQLiteErrorHandler.errorHandler(e);
            }
            ServletOutputStream out = response.getOutputStream();

            if(tmpColumna.toUpperCase().equals("BRETE_ADJUNTO")){

                out.write(obj.getBRETE_ADJUNTO());
            }

            if(tmpReq.equals("download")){
                if(obj.getBRETE_NOMBRE_ADJUNTO() != null){

```

```

        response.setHeader("Content-
Disposition","attachment;
filename="+obj.getBRETE_NOMBRE_ADJUNTO()+";");
response.setContentType("application/"+obj.getBRE
RETE_NOMBRE_ADJUNTO()+". "+obj.getBRETE_TIPO_ADJ
UNTO());
        out.flush();
        out.close();
    }
} else if(tmpReq.equals("ver")){
    if(obj.getBRETE_NOMBRE_ADJUNTO()
!= null){
        response.setHeader("Content-
Disposition","inline;
filename="+obj.getBRETE_NOMBRE_ADJUNTO()+";");
response.setContentType("application/"+obj.getBRE
TE_NOMBRE_ADJUNTO()+". "+obj.getBRETE_TIPO_ADJUNTO
());
        out.flush();
        out.close();
    }
}
} catch (FileNotFoundException error) {
error.printStackTrace();
} catch (IOException e) {
e.printStackTrace();
}
}
}
}

```

Clase de servicios web

```

package com.besixplus.sisac.ws;
import java.util.ArrayList;
import java.util.HashMap;
import java.io.Serializable;
import java.sql.Connection;
import java.sql.SQLException;
import javax.annotation.Resource;
import javax.jws.WebMethod;
import javax.jws.WebService;
import javax.xml.namespace.QName;
import javax.jws.soap.SOAPBinding;
import javax.jws.soap.SOAPBinding.Style;
import javax.xml.soap.SOAPException;
import javax.xml.soap.SOAPFactory;
import javax.xml.ws.soap.SOAPFaultException;
import javax.jws.WebParam;

```

```

import javax.servlet.http.HttpServletRequest;
import javax.xml.ws.WebServiceContext;
import javax.xml.ws.handler.MessageContext;
import com.besixplus.sisac.db.ManagerConnection;
import org.json.JSONArray;
import org.json.JSONObject;
import java.math.BigDecimal;

/**
 * CLASE Bsx_retencion_venta
 * CONTIENE LOS METODOS NECESARIOS PARA EL MANEJO DE LA
 * INFORMACION CONTENIDA EN LA TABLA BSX_RETENCION_VENTA POR
 * MEDIO DE SERVICIOS WEB.
 */
@WebService()
@SOAPBinding(style=Style.RPC)
public class Bsx_retencion_venta implements Serializable{

    private static final long serialVersionUID = 624860530;

    /**
     * SELECCIONA UN REGISTRO EN LA TABLA BSX_RETENCION_VENTA
     * EN UNA ESTRUCTURA JSON o XML.
     * @param inBrete_codigo Brete_codigo
     * @return String REGISTRO SELECCIONADO.
     */

    @WebMethod
    public String select(
        @WebParam(name="inBrete_codigo")String inBrete_codigo,
        @WebParam(name="format")String format
    ) throws SOAPException{
        com.besixplus.sisac.misc.Formatter tmpFormat = null;
        HttpServletRequest tmpRequest = (HttpServletRequest)
        wctx.getMessageContext().get(MessageContext.SERVLET_REQUEST);
        com.besixplus.sisac.objects.Bsx_retencion_venta
        tmpObj = new com.besixplus.sisac.objects.
        Bsx_retencion_venta();
        tmpObj.setBRETE_CODIGO(inBrete_codigo);
        StringBuilder outCadena = null;
        try{
            Connection con = ManagerConnection.getConnection();
            if(!com.besixplus.sisac.db.Bsx_objeto.isGrant(con,
            Thread.currentThread().getStackTrace()[1].getMethodName(
            ), Thread.currentThread().getStackTrace()[1].
            getClassName(),tmpRequest.getUserPrincipal().getName(),
            1)){
                con.close();
            }
            throw new
            SOAPFaultException(SOAPFactory.newInstance().createFault
            ("Acceso no autorizado.", new

```

```

        QName("http://schemas.xmlsoap.org/soap/envelope/", Thread
            .currentThread().getStackTrace()[1].getClassName()+"
            "+Thread.currentThread().getStackTrace()[1].getMethodNam
            e())));
    }
    con.setAutoCommit(!ManagerConnection.isDeployed());
    tmpObj = new com.besixplus.sisac.db.Bsx_retencion_
        venta(tmpObj).select(con);
    tmpFormat = new com.besixplus.sisac.misc.Formatter
        (format, tmpObj);
        outCadena = tmpFormat.getData();
        con.close();
    }catch(SQLException inException){

        com.besixplus.sisac.db.SQLiteErrorHandler.errorHandler(in
            Exception);
            throw new
                SOAPFaultException(SOAPFactory.newInstance().createFault
                    (inException.getMessage(), new
                        QName("http://schemas.xmlsoap.org/soap/envelope/",
                            Thread.currentThread().getStackTrace()[1].getClassName()
                                +
                                    "+Thread.currentThread().getStackTrace()[1].getMethodNam
                                    e()))));
    }catch (Exception inException){
        com.besixplus.sisac.db.SQLiteErrorHandler.errorHandler(inEx
            ception);
            throw new
                SOAPFaultException(SOAPFactory.newInstance().createFault
                    (inException.getMessage(), new
                        QName("http://schemas.xmlsoap.org/soap/envelope/",
                            Thread.currentThread().getStackTrace()[1].getClassName()
                                +
                                    "+Thread.currentThread().getStackTrace()[1].getMethodNam
                                    e()))));
    }
    if (tmpObj != null)
        return outCadena.toString();
    return null;
}
}
}

```

Funciones javaScript.

```

Function FrmBsx_retencion_venta(IN_SENTENCIA_BSX_
    RETENCION_VENTA,IN_RECORD_BSX_RETENCION_VENTA) {

    var btnGuardarBsx_retencion_venta = new Ext.Button({
        id:'btnGuardarBsx_retencion_venta',
        text:'Guardar',
        iconCls:'iconGuardar',
    });
}

```

```

        listeners:{
            click:function(){
if (!pnlBsx_retencion_venta.
getForm().isValid()){
                return;
            }
            var onSuccessOrFail = function(form, action) {
                var outResult = action.result;
                if (outResult == 'true') {
                    Ext.Msg.show({
                        title:tituloBsx_retencion_venta,
msg: 'La informaci\u00f3n ha sido almacenada.',
                        buttons: Ext.Msg.OK,
                        icon: Ext.MessageBox.INFO
                    });
                    winFrmBsx_retencion_venta.close();
                }
                else {
                    Ext.Msg.show({
                        title:tituloBsx_retencion_venta,
msg: 'La informaci\u00f3n no ha podido ser almacenada.',
                        buttons: Ext.Msg.OK,
                        icon: Ext.MessageBox.ERROR
                    });
                }
            }

            var tmpNombreAdjunto = null;
            var tipoAdjunto = null;
            if(filBrete_adjunto.getValue().length>0) {
                var tmpNombre=filBrete_adjunto
.getValue().split('.');
                tmpNombreAdjunto = tmpNombre[0];
                tipoAdjunto = tmpNombre[1];
            }

pnlBsx_retencion_venta.getForm().submit({
url: URL_WS+"Bsx_retencion_ventaSRV",
waitMsg: 'Enviando formulario...',
success:onSuccessOrFail,
failure:onSuccessOrFail,
params:{
request:IN_SENTENCIA_BSX_RETENCION_VENTA,
brete_codigo:txtBrete_codigo.getValue(),
bvent_codigo:CODIGO_VENTA,
brete_numero:txtBrete_numero.getValue(),
brete_fecha_reten:dtBrete_fecha_reten.getValue()?dtBrete_fecha_reten.getValue().format("Y-m-d"):null,
brete_total_reten:numBrete_total_reten.getValue(),
brete_nombre_adjunto:tmpNombreAdjunto,
brete_tipo_adjunto:tipoAdjunto,
brete_observacion:txtBrete_observacion.getValue().toUpperCase(),

```

```
retencion_detalle:grdBsx_retencion_venta_detalle.getStore().  
getJsonData()  
    }  
    });  
}  
  
}  
});  
  
}
```

6.8.5. Implantación

El módulo gestión de inventario y facturación es una parte del Sistema Administrativo Contable de la empresa BESIXLPLUS CIA. LTDA., es así que para su implantación es necesaria la previa implementación del módulo de Gestión de compras y proveedores, el módulo de gestión de clientes y seguridades y el módulo de contabilidad.

Cada uno de los módulos antes descritos están a cargo de un desarrollador, por tal motivo el grupo de desarrollo ha planificado y coordinado la metodología de desarrollo ha utilizar tanto en diseño de la interfaz, modelamiento de la base de datos, nomenclaturas y estándares de codificación.

Además ha utilizado un sistema de control de versiones Subversion que es un sistema centralizado para compartir información. En su núcleo está un repositorio, que es un almacén central de datos. El repositorio almacena información en forma de un árbol de ficheros - una jerarquía típica de ficheros y directorios. Cualquier número de clientes se conectan al repositorio, y luego leen o escriben esos ficheros. Al escribir datos, el cliente hace que la información esté disponible para los otros; al leer los datos, el cliente recibe la información de los demás. Tortoise SVN es un cliente para el sistema de control de versiones, mediante estas dos herramientas se ha podido trabajar en equipo y facilitará la integración de los módulos que forman el sistema.

Sin embargo se han realizado las pruebas necesarias del módulo de gestión de inventario y facturación utilizando servidores de prueba tanto en servidor de

aplicaciones, base de datos y reportes, comprobando la validez y fiabilidad de la información.

6.8.6. Pruebas

Las pruebas de software permiten verificar y revelar la calidad de un producto software. Son utilizadas para identificar posibles fallos de implementación, calidad, o usabilidad.

- Las pruebas no tienen el objeto de prevenir errores sino de detectarlos.

6.8.6.1. Pruebas unitarias

Verifica un componente o módulo del software. Se prueban caminos de control para detectar los errores.

Se realizaron pruebas de las funciones sql, clases java y código javaScript.

A continuación se presenta un ejemplo de prueba que se realizó sobre el método insert de la clase Bsx.retencion_venta del paquete com.Besixplus.sisac.db; utilizando JUnit herramienta open source que permite probar clases java de forma unitaria.

En la Figura 6.32 se puede observar un error en el envío de datos ya que existen dos parámetros con el mismo índice.

Al ejecutar la clase Bsx_retencion_ventaTest Figura 6.33 deberá retornar el **error generado** y la información no será almacenada.

Figura 6.32. Código de prueba - Método insert.

```
public String insert(
    java.sql.Connection inConnection
){
    String outResult = "true";
    try{
        CallableStatement stmInsert = inConnection.prepareCall("{ ? = call SCH_BSX.F_BSX_RETENCION_VENTA_INSERT
        (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?) }");
        stmInsert.registerOutParameter(1, java.sql.Types.VARCHAR);
        stmInsert.setString(2, this.getBsx_retencion_venta().getBRETE_CODIGO());
        stmInsert.setString(3, this.getBsx_retencion_venta().getBRETE_EVENTO());
        stmInsert.setString(4, this.getBsx_retencion_venta().getBRETE_NUMERO());
        stmInsert.setTimestamp(5, this.getBsx_retencion_venta().getBRETE_FECHA_RETEN() != null ? new
        java.sql.Timestamp(this.getBsx_retencion_venta().getBRETE_FECHA_RETEN().getTime()): null);
        stmInsert.setBigDecimal(6, this.getBsx_retencion_venta().getBRETE_TOTAL_RETEN());
        stmInsert.setBytes(7, this.getBsx_retencion_venta().getBRETE_ADJUNTO());
        stmInsert.setString(8, this.getBsx_retencion_venta().getBRETE_NOMBRE_ADJUNTO());
        stmInsert.setString(9, this.getBsx_retencion_venta().getBRETE_TIPO_ADJUNTO());
        stmInsert.setString(10, this.getBsx_retencion_venta().getBRETE_OBSERVACION());
        stmInsert.setBoolean(11, this.getBsx_retencion_venta().getBRETE_ESTADO());
        stmInsert.setString(12, this.getBsx_retencion_venta().getBRETE_USUARIO_INSERT());
        stmInsert.execute();
        this.getBsx_retencion_venta().setBRETE_CODIGO(stmInsert.getString(1));
        stmInsert.close();
    } catch (SQLException e) {
        com.besixplus.sisac.db.SQLExceptionHandler.errorHandler(e);
        outResult = e.getMessage();
    }
    return outResult;
}
```

Error en el índice del parámetro

Figura 6.33. Código de prueba – Clase Bsx_retencion_ventaTest

```
package com.besixplus.sisac.db;
import junit.framework.TestCase;
import java.math.BigDecimal;
import java.util.*;

public class Bsx_retencion_ventaTest extends TestCase {
    public final void testInsert() {
        String res = null;
        Date fecha = new Date();
        com.besixplus.sisac.objects.Bsx_retencion_venta tmpObjRetencion = new com.besixplus.sisac.objects.
        Bsx_retencion_venta();
        com.besixplus.sisac.db.ManagerConnection tmpConn = new com.besixplus.sisac.db.ManagerConnection();
        tmpObjRetencion.setBRETE_CODIGO("KEYGEN");
        tmpObjRetencion.setBRETE_EVENTO("EVENT1");
        tmpObjRetencion.setBRETE_NUMERO("12345467");
        tmpObjRetencion.setBRETE_FECHA_RETEN(fecha);
        tmpObjRetencion.setBRETE_TOTAL_RETEN(new BigDecimal(200));
        tmpObjRetencion.setBRETE_ADJUNTO(null);
        tmpObjRetencion.setBRETE_TIPO_ADJUNTO(null);
        tmpObjRetencion.setBRETE_NOMBRE_ADJUNTO(null);
        tmpObjRetencion.setBRETE_OBSERVACION("Ingreso de prueba");
        tmpObjRetencion.setBRETE_ESTADO(true);
        tmpObjRetencion.setBRETE_USUARIO_INSERT("anaF");
        res = new com.besixplus.sisac.db.Bsx_retencion_venta(tmpObjRetencion).insert(tmpConn.getConnection());
        System.out.println("Respuesta -> "+ res);
    }
}
```

Clase tester

JUnit Figura 6.34 reporta los fallos sobre el código ejecutado en la clase Bsx_retencion_ventaTest.

Al corregir los errores del método insert Figura 6.32 volvemos a ejecutar, la prueba unitaria se completa con éxito y sin fallos Figura 6.35.

Figura 6.34. JUnit – Reporte de fallos

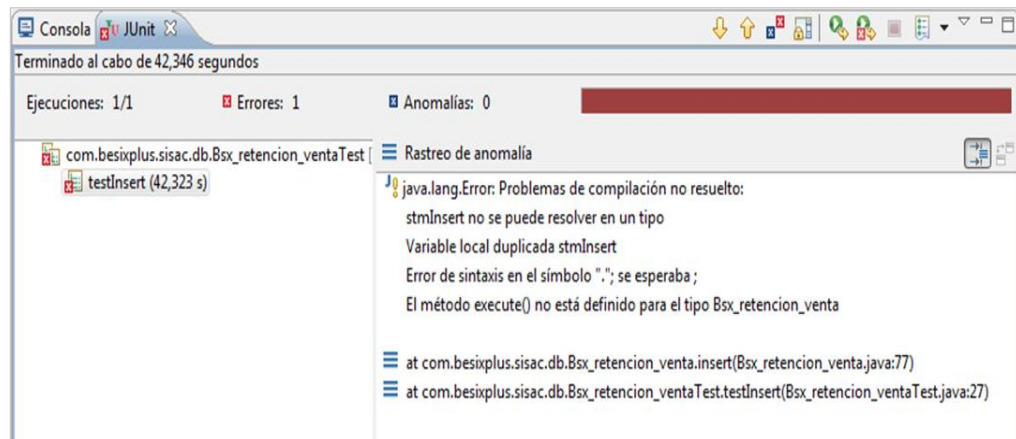
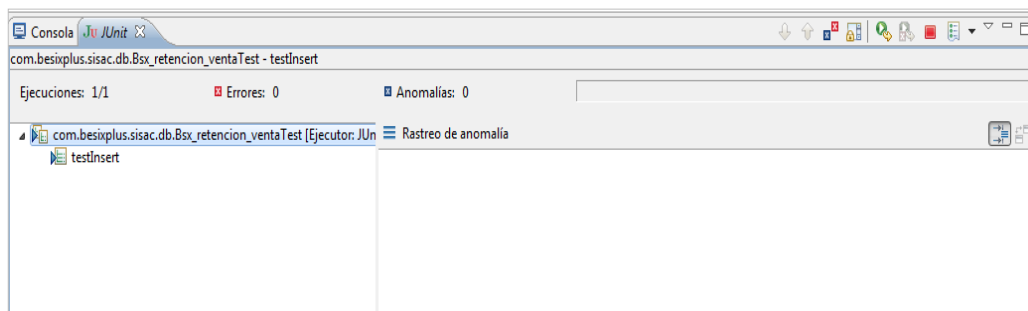


Figura 6.35. JUnit – Reporte de la prueba unitaria



6.8.6.2. Pruebas de caja blanca

Se basa en un minucioso exámen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que ejerciten conjuntos específicos de condiciones. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el estado deseado o Con este tipo de prueba de hizo énfasis en la revisión de los puntos que se describen a continuación.

- A través de la librería extjs se ha podido configurar los componentes para la validación de los datos utilizando máscaras o las opciones de configuración del componente utilizado.

- Se comprobó que las cantidades máximas de venta de un producto no superen a la cantidad existente en bodega.
- Se reviso que los cálculos de valores totales sean los corretos.
- Se valido la rutina de almacenamiento de datos relacionados, revisando que se almacenen de forma correcta en cada una de sus tablas en la base de datos.
- Se comprobó que los datos que se despliegan en los listados sean exactos.
- Se reviso los mensajes de alerta por posibles errores al ingreso o salida de información.

6.8.6.3. Pruebas de caja negra

Las pruebas de caja negra se centran en los requisitos funcionales del software, es decir, la prueba de caja negra permite obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. La prueba de caja negra intenta encontrar errores de las siguientes categorías:

- Funciones incorrectas o ausentes
- Errores de interfaz
- Errores de estructuras de datos
- Errores de rendimiento
- Errores de inicialización y de terminación

Con este tipo de prueba se hizo énfasis en la revisión de los puntos que se describen a continuación.

- Se verificó que la interfaz de usuario es amigable y de fácil uso.
- Se comprobó que el comportamiento de la interfaz de usuario es el correcto.

- Se comprobó que la interfaz de usuario es funcional en varios navegadores web.
- Se revisó y optimizó las funciones sql para mejorar el rendimiento en la selección y proceso de información.

6.9. CONCLUSIONES Y RECOMENDACIONES

6.9.1. Conclusiones

- La Empresa BESIXPLUS CIA. LTDA. en la actualidad cuenta con una herramienta que le permite administrar de manera óptima los procesos de inventario y facturación.
- Al tener al usuario final como parte del equipo de desarrollo, se pudo ir validando los procesos automatizados en cada una de las etapas, evitando así largas revisiones al final del desarrollo del módulo.
- Como lenguaje de programación se utilizó java y como IDE de desarrollo Eclipse herramienta que ha permitido desarrollar el proyecto de manera ordenada creando paquetes, clases y métodos, además cuenta con depuradores de código que ayudan a crear un programa limpio y libre de errores.
- La aplicación es 100% web tiene una interfaz similar al entorno de Windows XP, lo cual permite al usuario tener una experiencia muy parecida o igual a la que se tiene en las aplicaciones de escritorio, esto se ha logrado utilizando la librería ExtJS que permite crear aplicaciones RIA (Aplicaciones Ricas en Internet) mediante Javascript.
- Uno de los aspectos más importantes que se ha logrado con la implantación del módulo de gestión de inventario y facturación en la Empresa es optimizar tiempos y costos en la entrega de productos mejorando la calidad de servicio al cliente, además la información se encuentra organizada, segura y disponible para los usuarios del sistema.

6.9.2. Recomendaciones

- Desarrollar las aplicaciones en base a los requerimientos del usuario para así obtener un producto de calidad, óptimo y seguro que permita resolver los problemas que conlleva el manejo manual de la información.
- Mantener una buena comunicación con los usuarios del sistema y directivos de la empresa o institución donde se está implantado la aplicación para así conocer sus inquietudes y necesidades.
- Las personas que utilicen la aplicación deben tener una adecuada y amplia capacitación, para que así puedan conocer y comprender las diferentes funcionalidades del sistema y evitar un mal manejo o ingreso de datos erróneos.
- Se sugiere al Administrador del Sistema tener especial cuidado en el manejo de parámetros generales que utiliza la aplicación ya que un dato erróneo o faltante, puede originar información inconsistente.
- La seguridad es uno de los aspectos más importantes en una aplicación por lo que es recomendable que el Administrador del Sistema defina los privilegios y roles que tendrá cada usuario del sistema.
- Realizar respaldos de la base de datos continuamente de esta manera la información se encuentra segura y podrá ser recuperada si llega a surgir algún imprevisto con el servidor de base de datos.

6.10. GLOSARIO DE TÉRMINOS

AJAX: (*Asynchronous JavaScript And XML*), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA.

API: (*Application Programming Interface*) Interfaz de programación de aplicaciones es el conjunto de funciones y procedimientos o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas (también denominadas comúnmente "librerías").

CDDL: (*Common Development and Distribution License*) Licencia Común de Desarrollo y Distribución es una licencia de código abierto (OSI) y libre, producida por Sun Microsystems, basada en la Mozilla Public License o MPL, versión 1.1.

CSS: (*Cascading Style Sheets*) es un lenguaje de hojas de estilo para describir la semántica de la presentación (el aspecto y el formato) de un documento escrito en un lenguaje de marcas

DBMS: (*Database Management System*) El sistema de gestión de bases de datos es un paquete de software con los programas informáticos que controlan la creación, el mantenimiento y el uso de una base de datos.

DDL: (*Data Definition Language*) Lenguaje de descripción de datos es un lenguaje de programación para definir estructuras de datos.

DEBUGGEAR: (*Depurar*) Seguir paso a paso cada instrucción ejecutada para así localizar bugs o errores en el código.

DML: (*Data Manipulation Language*) Lenguaje de Manipulación de datos es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los usuarios de la misma llevar a cabo las tareas de consulta o manipulación de los datos, organizados por el modelo de datos adecuado.

DOM: (*Document Object Model*) 'Modelo de Objetos del Documento' o 'Modelo en Objetos para la representación de Documentos' es esencialmente una interfaz de programación de aplicaciones (API) que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos.

FIREBUG: es una extensión de Firefox creada y diseñada especialmente para desarrolladores y programadores web. Es un paquete de utilidades con el que se puede analizar (revisar velocidad de carga, estructura DOM), editar, monitorizar y depurar el código fuente, CSS, HTML y JavaScript de una página web de manera instantánea e *inline*.

GNU GPL: (*GNU General Public License*) Es una licencia creada por la Free Software Foundation en 1989 (la primera versión), y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

HTML: (*HiperText Markup Language*) Lenguaje de Marcación de Hipertexto, es un lenguaje se utiliza comúnmente para establecer la estructura y contenido de un sitio web, tanto de texto, objetos e imágenes.

IDE: (*Integrated Development Environment*) Entorno de desarrollo integrado es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un sólo lenguaje de programación o bien, poder utilizarse para varios.

JDK: (*Java Development Kit*) Es un software que provee herramientas de desarrollo para la creación de programas en Java.

jQuery: Es una biblioteca de JavaScript, permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar

eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

OPEN SOURCE: Código abierto es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones éticas y morales las cuales destacan en el llamado software libre.

RIA: (*Rich Internet Applications*) Aplicaciones de Internet enriquecidas, son aplicaciones web que tienen la mayoría de las características de las aplicaciones de escritorio tradicionales. Estas aplicaciones utilizan un navegador web estandarizado para ejecutarse y por medio de complementos o mediante una máquina virtual se agregan las características adicionales.

SOFTWARE: Se conoce como software al equipamiento lógico o soporte lógico de un sistema informático; comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos, que son llamados hardware.

SQL: (*Structured Query Language*) Lenguaje de consulta estructurado es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas.

UDDI: (*Universal Description, Discovery and Integration*) Es uno de los estándares básicos de los servicios Web cuyo objetivo es ser accedido por los mensajes SOAP y dar paso a documentos WSDL, en los que se describen los requisitos del protocolo y los formatos del mensaje solicitado para interactuar con los servicios Web del catálogo de registros.

UML: (*Unified Modeling Language*) Lenguaje Unificado de Modelado, es un lenguaje gráfico para especificar, visualizar, construir y documentar los sistemas de software, representa un conjunto de las mejores prácticas que han probado ser exitosas en el modelado de sistemas grandes y complejos.

WEB: El sistema de documentos (o páginas web) interconectados por enlaces de hipertexto, disponibles en Internet.

WSDL: (*Web Services Description Language*) Formato XML que se utiliza para describir servicios Web

W3C: (*World Wide Web Consortium*) Es un consorcio internacional que produce recomendaciones para la World Wide Web.

6.11. BIBLIOGRAFÍA

6.11.1. Bibliografía de Libros

Caudwell, P. Servicios Web XML. Editorial ANAYA, 2003.

Chiavenato, Idalberto. Administracion, teórica, proceso y práctica. Colombia, McGraw - Hill Interamericaba, S.A.; 2011. p.3

Terán Gandarillas, Gonzalo J. Temas de contabilidad basica e intermedia. 4ta Edición; Editorial Educación y Cultura; Pág 4.

6.11.2. Bibliografía de Internet

La información fue consultada entre Julio y Noviembre del 2011

Casanovas, Josep . Usabilidad y arquitectura del software. (31/07/2004). Obtenido de: http://www.alzado.org/articulo.php?id_art=355

Wikipedia. Arquitectura monolítica. (s.f.). Obtenido de:
[http://es.wikipedia.org/w/index.php?title=Arquitectura_monol%C3%ADtica_\(inform%C3%A1tica\)&action=edit&redlink=1](http://es.wikipedia.org/w/index.php?title=Arquitectura_monol%C3%ADtica_(inform%C3%A1tica)&action=edit&redlink=1)

Wikipedia. Cliente-servidor. (s.f.). Obtenido de:
<http://es.wikipedia.org/wiki/Cliente-servidor>

Wikipedia. Programación por capas. (s.f.). Obtenido de:
http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles

Wikipedia. Arquitectura orientada a servicios. (s.f.). Obtenido de:
http://es.wikipedia.org/wiki/Arquitectura_orientada_a_servicios

PostgreSQL. PostgreSQL 8.1.23 Documentación. (s.f.). Obtenido de:
<http://www.postgresql.org/docs/8.1/static/preface.html#INTRO-WHATIS>

PostgreSQL. PostgrSQL. (s.f.). Obtenido de: <http://www.postgresql.org/about/>

Romer, Diego F. Servidor de Aplicaciones: ¿Qué es? (s.f.). Obtenido de:
<http://www.editum.org/Servidor-De-Aplicaciones-Que-Es-p-604.html>

Alfárez Sánchez, José A. Instalación, configuración y administración del servidor de aplicaciones JBoss. (s.f.). Obtenido de:
<http://es.scribd.com/doc/58116358/Jboss>

JasperSoft BI. (s.f.). Obtenido de:
http://www.ptb.com.mx/index.php?option=com_content&view=article&id=44&Itemid=54

Clavel Rosales, Evelio; Membrives Rodríguez, Dariel. Subsistema de gestión de cartas de créditos para el sistema Automatizado de gestión bancaria (SAGEB). (09/2010.). Obtenido de:
<http://semanatecnologica.fordes.co.cu/ocs-2.3.2/public/site/165.pdf>

Pavón Molina, Francisco Javier. Principles of programming II Object Oriented Programming. (03/2006). Obtenido de:
<http://www.aiu.edu/applications/DocumentLibraryManager/upload/Principles%20of%20Programming%20II.pdf>

IBM Global Business Services. Arquitectura orientada a servicios. (10/2006). Obtenido de: <http://www-304.ibm.com/easyaccess/fileserv?contentid=129573>

Principios SOA. Publicado por Trumbulianos IV 2007 En:
<http://arquiware.blogspot.com/>

W3C. Simple Object Access Protocol (SOAP) 1.1. (08//03/2000). Obtenido de:
<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

Ventajas de los servicios web. [Imagen]. (2010). Obtenido de:
<http://agronomiaunlu.blogspot.com/>

García, Mayer Horna; Lara Vazquez, Jonathan. Aplicaciones AJAX. (s.f.). Obtenido de: <http://www.slideshare.net/njelov/ajax-huancayo>

Wikipedia. HTML. (s.f.). Obtenido de: <http://es.wikipedia.org/wiki/HTML>

Wikipedia. HTML dinámico. (s.f.). Obtenido de:
http://es.wikipedia.org/wiki/HTML_din%C3%A1mico

Wikipedia. XML. (s.f.). Obtenido de: <http://www.w3.org/XML/>

Wikipedia. Cascading Style Sheets. (s.f.). Obtenido de:
http://en.wikipedia.org/wiki/Cascading_Style_Sheets

Wikipedia. Document Object Model. (s.f.). Obtenido de:
http://es.wikipedia.org/wiki/Document_Object_Model

Wikipedia. Ext JS. (s.f.). Obtenido de: http://es.wikipedia.org/wiki/Ext_JS

Wikipedia. Navegador web. (s.f.). Obtenido de:
http://es.wikipedia.org/wiki/Navegador_web

González, Mayra. Guerra de navegadores. [Imagen]. (2011). Obtenido de
<http://www.solo-opiniones.com/2011/03/guerra-de-navegadores/>

Gómez, Geovanny E. Características y sistemas de valuación de inventarios de mercancías. (2001). Obtenido de :
<http://www.gestipolis.com/canales/financiera/articulos/14/inventarios.htm>

Wikipedia. Factura. (s.f.). Obtenido de: <http://es.wikipedia.org/wiki/Factura>

Wikipedia. Administración. (s.f.). Obtenido de:
<http://es.wikipedia.org/wiki/Administraci%C3%B3n>

Acuariano. Introducción a SOA. Obtenido de: http://www.epidata consulting.com/tikiwiki/tiki-read_article.php?articleId=44

Tentor, Julio. Arquitectura de N-Capas y N-Niveles. [Imagen]. Obtenido de:
<http://www.jtentor.com.ar/post/Arquitectura-de-N-Capas-y-N-Niveles.aspx>

Anexos



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

Entrevistado: _____

Fecha : _____

Nota: la información que se recopila es para uso exclusivo del trabajo de investigación a efectuarse en la empresa BESIXPLUS CIA LTDA. por lo tanto se recomienda que la información sea verídica.

Cuestionario

1. ¿Qué Sistema de Inventario utiliza para el control de movimientos de la mercadería?

Sistema de Inventario Permanente (kardex)

Sistema de Inventario Perpetuo

2. ¿Cómo califica el procedimiento utilizado en la Empresa para el control de mercadería?

Malo

Regular

Bueno

Excelente

3. ¿Conoce usted el procedimiento para elaborar una factura?

Si

No

4. ¿Conoce usted las disposiciones reglamentarias emitidas por el SRI relacionadas a los comprobantes de venta?

Si

No

5. ¿Cuáles piensa usted que serían las ventajas de utilizar el sistema de inventarios y facturación a través de internet?

.....
.....
.....

6. ¿Cómo califica usted el servicio al cliente?

Malo

Regular

Bueno

Excelente

7. ¿La información de inventarios se encuentra almacenada en algún tipo de base de datos?

Base de datos manual

Base de datos semi-automatizada

Base de datos automatizada

Ninguna

8. ¿Cuál de estos paquetes contables ha manejado usted?

Mónica

Fenix

MicroXsystem

Ninguno

Otros

9. ¿Existe en su Empresa un manual de funciones?

Si

No

10. ¿Cómo califica el crecimiento en ventas de su Empresa?

Malo

Regular

Bueno

Excelente

Entrevistado



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

Presupuesto del proyecto

Nº	Recurso	Cantidad	V. unitario	Total
1	Computador	650 Horas	0.80	520.00
2	Internet	360 Horas	0.80	288.00
3	Impresora	1	60.00	60.00
4	Tinta para impresora	1	7.00	7.00
5	Hojas de papel bon	1	5.00	5.00
6	Memoria Flash USB	1	-	10.00
7	Anillados	Global	1.00	1.00
8	Empastado	3	30	90.00
9	Carpetas	3	0.50	1.50
10	Material de Oficina	Global	-	5.00
11	Transporte	Global	-	240.00
12	Alimentación	Global	-	240.00
13	Servicios Básicos	Global	-	150.00
14	Talento humano	-	-	800.00
			Total	2417.50

BESIXPLUS CIA. LTDA.



MÓDULO DE GESTIÓN DE INVENTARIO Y FACTURACIÓN MANUAL DE USUARIO

Ana Flores

Ambato – Ecuador
2011

Este manual tiene como objetivo guiar al usuario en la operación del software mediante descripciones, capturas de pantallas orientadas a los casos de uso con el fin de facilitar el entendimiento del mismo.

ENTORNO DE LA APLICACIÓN

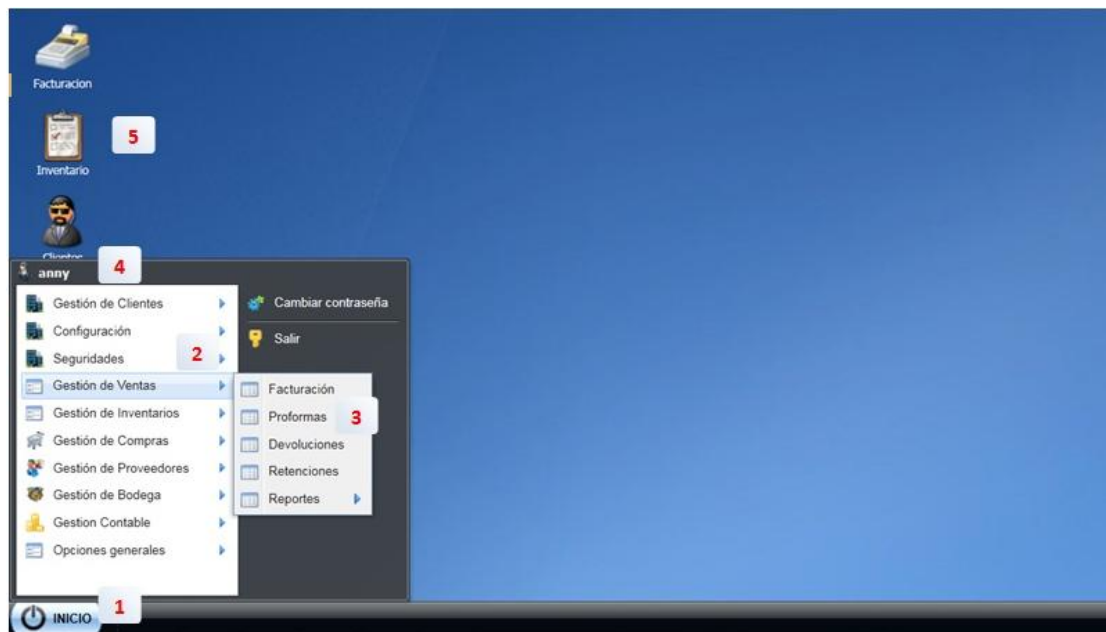


Figura 1. Pantalla de entorno de la aplicación

La aplicación presenta un entorno similar a Windows XP donde podemos observar:

- 1.- Menú inicio y barra de tareas
- 2.- Menú de módulos que forman el sistema.
- 3.- Opciones de cada uno de los módulos del sistema.
- 4.- Información de usuario que ingreso al sistema.
- 5.- Accesos directos a los módulos del sistema.

ADMINISTRACIÓN DE VENTAS



FACTURACIÓN

En esta parte se explicará acerca de la administración de la información relacionada con la facturación de productos y/o servicios que ofrece la empresa.

Acceso a la opción de Facturación

Para ingresar a esta pantalla podemos hacerlo desde el acceso directo en el escritorio de la aplicación seleccionando la opción “Facturación” o por el menú Inicio eligiendo la opción Gestión de ventas “Facturación”, al seleccionar la opción se visualizará el formulario con el listado de facturas registradas Figura1A.

Descripción del formulario

En el formulario Listado de facturas tenemos las siguientes secciones:

CI/RUC	Cliente	Nº comprobante	Fecha emision	Total venta	Forma de pago	Vendedor
1801051150	VIRGINIA PIMBO CHINGO	000030	11/10/2011	\$79.54	CONTADO	anny
1704176716	GERARDO FLORES LOPEZ	000029	09/10/2011	\$40.20	CONTADO	damian
1804192837001	GABRIELA MARANON BURBANO	000028	09/10/2011	\$30.00	CONTADO	anny
1801051150001	VERONICA TOAZA PACHA	000027	09/10/2011	\$90.00	CONTADO	damian
1804192837	ANA FLORES PIMBO	000026	08/10/2011	\$20.80	CONTADO	damian
0000000000	CONSUMIDOR FINAL	000025	08/10/2011	\$15.98	CONTADO	william
1804192837	ANA FLORES PIMBO	000024	08/10/2011	\$30.00	CONTADO	damian
0000000000	CONSUMIDOR FINAL	000023	07/10/2011	\$120.00	CONTADO	lucho
1804192837	ANA FLORES PIMBO	000022	06/10/2011	\$79.54	CONTADO	damian
1801051150	VIRGINIA PIMBO CHINGO	000021	06/10/2011	\$45.70	CONTADO	damian
1801051150001	VERONICA TOAZA PACHA	000019	04/10/2011	\$54.00	CONTADO	damian
1801051150	VIRGINIA PIMBO CHINGO	000018	04/10/2011	\$79.54	CONTADO	lucho
1804192837	ANA FLORES PIMBO	000017	01/10/2011	\$30.00	CONTADO	damian
0000000000	CONSUMIDOR FINAL	000016	01/10/2011	\$79.54	CONTADO	damian
1804192837	ANA FLORES PIMBO	000015	01/10/2011	\$20.94	CONTADO	anny

Figura 1A. Listado de facturas

- 1.- Encabezado.- en este se encuentra el titulo y una descripción del formulario.
- 2.- Búsqueda.- permite realizar búsquedas inteligentes de los datos.

3.- Datos.- contiene la lista de facturas emitidas los datos pueden ser ordenados ascendente y descendentemente por los distintos campos del listado.

4.- Paginación.- Permite mostrar una lista larga de datos en varias páginas

5.- Opciones del formulario.- Muestra las diferentes tareas a realizar en el formulario actual:

Nuevo.- Muestra el formulario para el ingreso de una nueva Factura.

Detallar.- Muestra el detalle de la factura.

Eliminar.- Permite eliminar un dato seleccionado de la lista.




Salir.- Cierra la ventana activa.

Registro de facturas

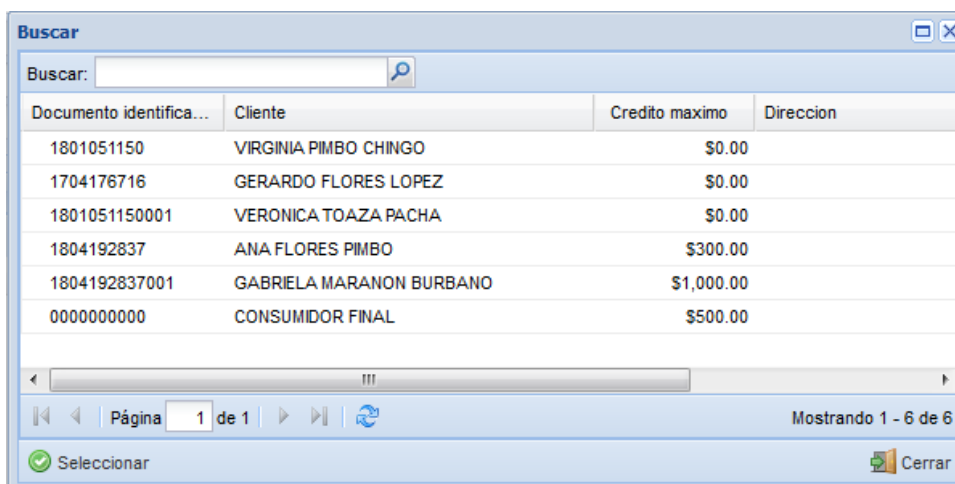
Al seleccionar la opción **Nuevo** del formulario Listado de facturas Figura 1A se visualiza el siguiente formulario Figura 2A, donde debemos llenar los campos requeridos para ingresar una nueva factura.

Código	Descripción	Cantidad	% Descu...	V. Unitario	Total
--------	-------------	----------	------------	-------------	-------

Figura 2A. Ingreso nueva factura

Dentro del formulario Figura 2A al dar click en el botón Buscar  podemos visualizar el Listado de Clientes Figura 3A; para agregar productos al detalle de la Factura presionamos el botón  Añadir el cual visualizara el Listado de Productos y servicios Figura 4A y para eliminar un ítem del detalle de la Factura seleccionamos el dato y luego presionamos el botón Eliminar .

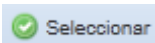
Al guardar la información se genera automáticamente el reporte de la factura en formato pdf para poder imprimirla.



Documento identifica...	Cliente	Credito maximo	Direccion
1801051150	VIRGINIA PIMBO CHINGO	\$0.00	
1704176716	GERARDO FLORES LOPEZ	\$0.00	
1801051150001	VERONICA TOAZA PACHA	\$0.00	
1804192837	ANA FLORES PIMBO	\$300.00	
1804192837001	GABRIELA MARANON BURBANO	\$1,000.00	
0000000000	CONSUMIDOR FINAL	\$500.00	

Figura 3A. Listado de Clientes

En el Listado de Clientes Figura 3A, se debe marcar el Cliente y dar click en el botón



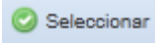
; o dar doble click en el ítem seleccionado.


Código	Nombre	Característica	Marca	Color	% Desc...	Precio	Existencias	Unidad
Ubicación: BODEGA NORTE								
<input type="checkbox"/>	TCL-001	TECLADO	GENIUS	NEGRO	0.0	\$39.00	36	UNIDAD
<input type="checkbox"/>	MS-001	MOUSE	HHH	TTTT	0.0	\$52.00	5	UNIDAD
<input type="checkbox"/>	CPU-001	CPU	HHH	HHHH	2.0	\$1,152.00	46	UNIDAD
Ubicación: SERVICIOS								
<input type="checkbox"/>	FORMT_001	FORMATEO DE DISCO DURO	SSDFSDF		0.0	\$10.40	0	UNIDAD

Mostrando 1 - 4 de 4

Seleccionar Cerrar

Figura 4A. Listado de Productos y Servicios

En el listado de productos y servicios Figura 4A, se debe marcar los productos y/o servicios deseados y dar click en el botón  y estos se añadirán al detalle de la Factura.

Además se debe considerar que si un producto maneja **Series** se deberá seleccionar las series a facturar. En el detalle de la factura Figura 5A se visualizará un botón  y se mostrará el Listado de Series del producto Figura 6A.


Detalle							
<input type="button" value="Añadir"/> <input type="button" value="Eliminar"/>							
	Código	Descripción		Cantidad	% Descu...	V. Unitario	Total
1	TCL-001	TECLADO		1	0.0	\$39.00	\$39.00
2	MS-001	MOUSE		0	0.0	\$52.00	\$0.00
3	CPU-001	CPU		1	2.0	\$1,128.96	\$1,128.96

Figura 5A. Detalle de la factura

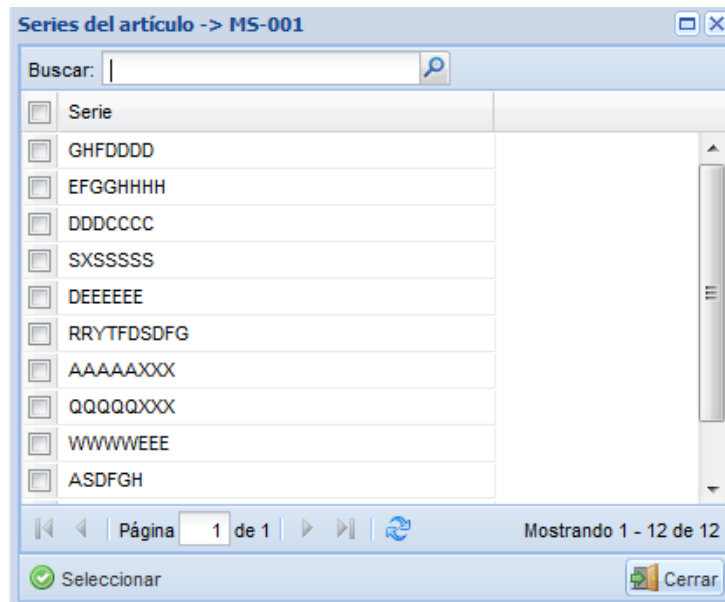
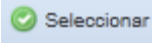



Figura 6A. Listado de series del producto seleccionado

En el Listado de series Figura 6A debemos seleccionar las series a facturar y dar click en el botón .

Detallar factura

Para detallar una Factura ya almacenada se debe seleccionar un ítem del Listado de Facturas Figura 1A y presionar el botón Detallar  de la Figura 1A o dar doble click sobre el registro seleccionado; se visualizará el formulario Figura 7A con los datos de la Factura seleccionada en modo de lectura ya que no se puede hacer ningún cambio.

Datos del cliente

CI/Ruc: 1804192837 Limite de crédito: 0
 Cliente: ANA FLORES PIMBO
 Observación: VENTA DE PRUEBA1
 Mot. anulación:

Datos del comprobante de venta

Nº: VENT-2011-1
 Fecha emisión: 12/11/2011
 Vendedor: anny
 Ingr. retención:

Detalle

Código	Descripción	Cantidad	% Descu...	V. Unitario	Total
1	TCL-001 TECLADO	1	0.0	\$39.00	\$39.00
2	CPU-001 CPU	1	2.0	\$1,128.96	\$1,128.96
3	FORMT_001 FORMATEO DE DISCO DURO	1	0.0	\$10.40	\$10.40
4	MS-001 MOUSE	3	0.0	\$52.00	\$156.00

Valores a cancelar

Subtotal: 1334.36
 Descuento 0 %: 0
 Iva 12 %: 160.12
TOTAL A PAGAR: 1494.48

Formas de pago

A Contado

Pago	Valor	Nº Transf. o che...	Ba
EFFECTIVO	\$1,494.48		


A Crédito

Valor:
 Tipo cuota: Seleccione el tipo de cu...
 Plazo credito: Seleccione el plazo de c...
 Observación:

Guardar Cancelar Cerrar

Figura 7A. Detalle de la Factura

Eliminación de facturas

Para eliminar una Factura del listado Figura 1A, debemos seleccionar la factura y presionar el botón Eliminar  del formulario Figura 1A. Se visualizará el siguiente mensaje Figura 8A para confirmar la eliminación del registro.

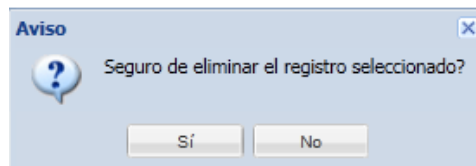


Figura 8A. Mensaje de verificación de Eliminación de la Factura

PROFORMAS

En esta parte se explicará acerca de la administración de la información relacionada con las Proformas de productos y/o servicios.

Acceso a la opción de Proformas

Para ingresar a esta pantalla podemos hacerlo desde el menú Inicio eligiendo la opción Gestión de ventas “Proformas”.

En el formulario Figura 1B podemos visualizar la información de las proformas registradas.

Número	CI/Ruc	Cliente	Fecha emision	Fecha vencim	Vendedor
000001	1801051150	VIRGINIA PIMBO CHINGO	26/10/2011	26/10/2011	anny
000003	1801051158	ESTELA CASTELLANO RUIZ	26/10/2011	26/10/2011	anny
000002	1801051150	VIRGINIA PIMBO CHINGO	03/11/2011	13/11/2011	anny
000004	1804192839	OMAR GAVILANEZ POZO	03/11/2011	10/11/2011	anny

Figura 1B. Listado de Proformas

Registro de proformas

Al seleccionar la opción **Nuevo** del formulario Listado de Proformas Figura 1B se visualiza el siguiente formulario Figura 2B, donde debemos llenar los campos requeridos para ingresar una nueva Proforma.

Proforma

Proforma
El formulario permite administrar la información de la tabla Proforma

Datos del cliente

CI/Ruc: Parroquia:

Nombres: Calle:

Apellidos: Telefono:

Datos generales

Numero:

Fecha emision:

Fecha vencim:

Vendedor:

Detalle

+ Añadir - Eliminar

Código	Descripción	Unidad	Cantidad	V. Unitario	Total	Observación
--------	-------------	--------	----------	-------------	-------	-------------

Observación:




Subtotal:

%Iva:

Total:

Guardar Cancelar Cerrar

Figura 2B. Ingreso nueva Proforma

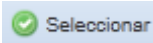
Dentro del formulario Figura 2B al dar click en el botón Buscar  podemos visualizar el Listado de Clientes Figura 3B, para agregar productos al detalle de la Proforma presionamos el botón Añadir  el cual visualizará el Listado de Productos y servicios Figura 4B y para eliminar un ítem del detalle de la Proforma seleccionamos un ítem y luego presionamos el botón Eliminar .

Al guardar la información se genera automáticamente el reporte de la proforma en formato pdf para poder imprimirla.

Documento identifica...	Cliente	Credito maximo	Direccion
1801051150	VIRGINIA PIMBO CHINGO	\$0.00	
1704176716	GERARDO FLORES LOPEZ	\$0.00	
1801051150001	VERONICA TOAZA PACHA	\$0.00	
1804192837	ANA FLORES PIMBO	\$300.00	
1804192837001	GABRIELA MARANON BURBANO	\$1,000.00	
0000000000	CONSUMIDOR FINAL	\$500.00	

Figura 3B. Listado de clientes

En el Listado de Clientes Figura 3B, se debe marcar el cliente y dar click en el botón

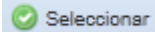


; o dar doble click en el ítem seleccionado.


Código	Nombre	Característica	Marca	Color	% Descuento	Precio	Existencias	Unidad
Ubicación: BODEGA NORTE								
ART-FLA-001	FLASH MEMORY	4G	KINGSTON	NEGRO, RO	5.0	\$18.70	42	UNIDAD
ART-MON-001	MONITOR	PLASMA	LG	NEGRO	2.0	\$630.00	29	UNIDAD
ART-TEC-001	TECLADO	TECLADO ESPAÑOL	GENIUS	NEGRO	1.0	\$17.44	9	UNIDAD
ART-RAT-001	RATON	USB	GENIUS	NEGRO	2.0	\$7.49	7	UNIDAD

Figura 4B. Listado de Productos y Servicios

En el Listado de Productos y Servicios Figura 4B, se debe marcar los productos

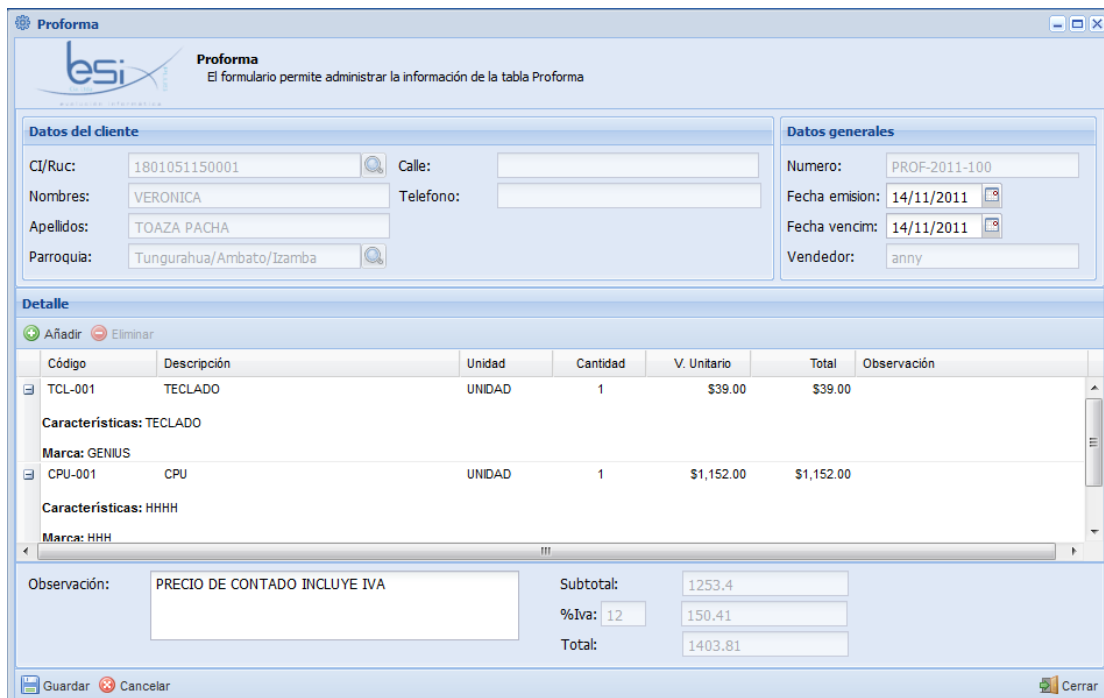
deseados y dar click en el botón 

Edición de proformas

Para Editar una proforma almacenada debemos seleccionar un ítem del Listado de Proformas Figura 1B, luego presionar el botón Editar  de la Figura 1B o dar doble click sobre el registro seleccionado; se visualizará el formulario Figura 5B.

No se podrá editar los siguientes campos:

- CI/Ruc
- Nombres
- Apellidos
- Parroquia
- Calle
- Teléfono
- Número
- Vendedor



Proforma
El formulario permite administrar la información de la tabla Proforma

Datos del cliente

CI/Ruc: 1801051150001 Calle:
Nombres: VERONICA Telefono:
Apellidos: TOAZA PACHA
Parroquia: Tungurahua/Ambato/Izamba

Datos generales

Numero: PROF-2011-100
Fecha emision: 14/11/2011
Fecha vencim: 14/11/2011
Vendedor: anny

Detalle

Añadir Eliminar

Código	Descripción	Unidad	Cantidad	V. Unitario	Total	Observación
TCL-001	TECLADO	UNIDAD	1	\$39.00	\$39.00	
Características: TECLADO						
Marca: GENIUS						
CPU-001	CPU	UNIDAD	1	\$1,152.00	\$1,152.00	
Características: HHHH						
Marca: HHH						


Observación: PRECIO DE CONTADO INCLUYE IVA

Subtotal: 1253.4
%Iva: 12 150.41
Total: 1403.81

Guardar Cancelar Cerrar

Figura 5B. Edición de la proforma

Eliminación de proformas

Para eliminar una Proforma del listado Figura 1B, debemos seleccionar la proforma a eliminar y presionar el botón Eliminar  del formulario Figura 1B. Se visualizará el siguiente mensaje Figura 6B para confirmar la eliminación del registro.

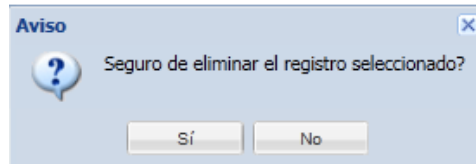


Figura 6B. Mensaje de verificación de Eliminación de la Proforma



RETENCIONES

En esta parte se explicará acerca de la administración de la información relacionada con las retenciones a las ventas realizadas.

Acceso a la opción de Retenciones

Para ingresar a esta pantalla podemos hacerlo desde el menú Inicio eligiendo la opción Gestión de ventas “Retenciones”.

En el formulario Figura 1C podemos visualizar la información de las retenciones registradas.

Nº de retención	Nº comprobante de venta	Fecha retención	Total retención
000904	000001	03/10/2011	4.0




Figura 1C. Listado de Retenciones


Registro de retenciones

Al seleccionar la opción **Nuevo** del formulario Listado de Retenciones Figura 1C se visualiza el siguiente formulario Figura 2C, donde debemos llenar los campos requeridos para ingresar una nueva Retención.

Ejercicio fiscal	Base imponible para la retención	Impuesto	% de retención	Valor retenido
------------------	----------------------------------	----------	----------------	----------------

Figura 2C. Ingreso nueva retención

Dentro del formulario Figura 2C al dar click en el botón Buscar  podemos visualizar el Listado de Facturas Figura 3C, para agregar impuestos al detalle de la retención presionamos el botón Añadir  el cual añadirá una fila al detalle de la Retención para digitar los valores del impuesto; y para eliminar un ítem del detalle de la retención seleccionamos el ítem y luego presionamos el botón Eliminar .

Además se puede adjuntar archivos presionando el botón . Al guardar la información se visualizara en el listado Figura 1C.

Buscar

Buscar:

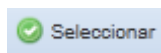
N°	CVRUC	Cliente	Fecha emision	Total venta	Forma de pago
VENT-2011-4	1801051150	VIRGINIA PIMBO CHINGO	12/11/2011	\$1,350.44	CONTADO
VENT-2011-2	1804192837001	GABRIELA MARANON BURBANO	12/11/2011	\$1,378.00	CREDITO
VENT-2011-1	1804192837	ANA FLORES PIMBO	12/11/2011	\$1,494.48	CONTADO

Mostrando 1 - 3 de 3

Seleccionar Cerrar


Figura 3C. Listado de Facturas

En el listado de facturas Figura 3C, se debe marcar la factura y dar click en el botón



; o dar doble click en el ítem seleccionado.

Edición de retenciones

Para editar los datos de una Retención almacenada debemos seleccionar un ítem del Listado de Retenciones Figura 1C y presionar el botón Editar  de la Figura 1C, o dar doble click en el ítem seleccionado; se visualiza el formulario Figura 4C.

No se podrá editar los siguientes campos:

- N° de comprobante de venta
- Total retención
- O el detalle de la retención ingresada.

Retenciones
El formulario permite administrar la información de la tabla Retenciones

Nº Comprobante de venta: 000001
 Nº de retención: 000904
 Fecha reten: 03/10/2011

Observación: RETENCION DE BSX
 Adjunto: graficos y chi.xlsx
 Total retención: 4

Impuestos


+ Añadir - Eliminar

Ejercicio fiscal	Base imponible para la retención	Impuesto	% de retención	Valor retenido
2011	\$1.00	IVA	0.0	\$0.00

Guardar Cancelar Cerrar

Figura 4C. Edición de la retención

Eliminación de retenciones

Para eliminar una Retención del listado Figura 1C, debemos seleccionar la retención a eliminar y presionar el botón Eliminar  del formulario Figura 1C. Se visualizará el siguiente mensaje Figura 5C para confirmar la eliminación del registro.

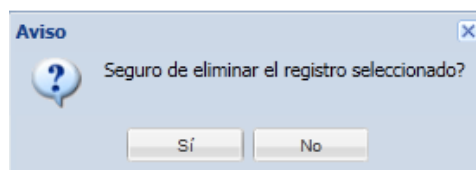


Figura 5C. Mensaje de verificación de Eliminación de la retención

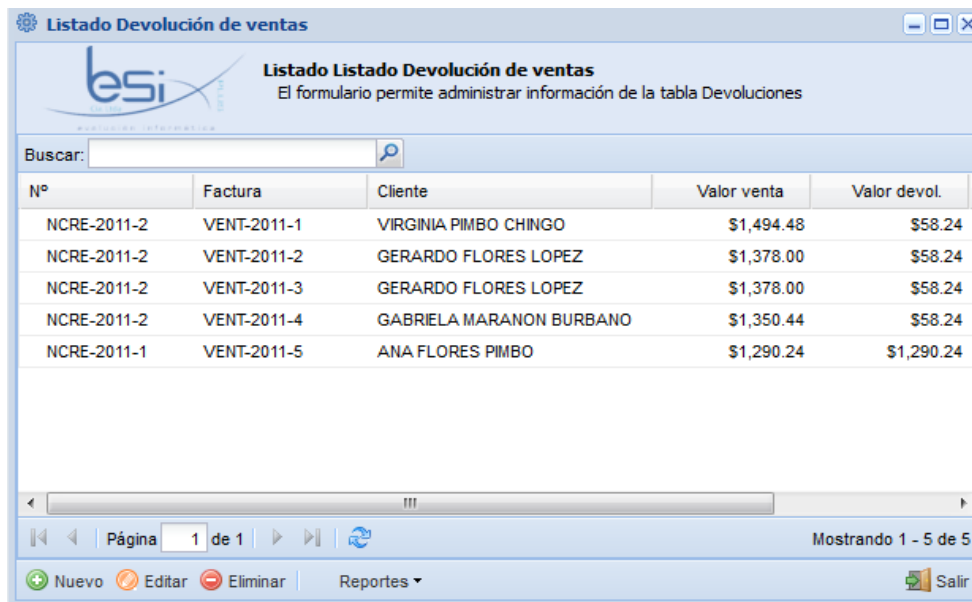
DEVOLUCIONES

En esta parte se explicará acerca de la administración de la información relacionada con las devoluciones de las ventas.

Acceso a la opción de devoluciones

Para ingresar a esta pantalla podemos hacerlo desde el menú Inicio eligiendo la opción Gestión de ventas “Devoluciones”.

En el formulario Figura 1D podemos visualizar la información de las Devoluciones registradas.






Nº	Factura	Cliente	Valor venta	Valor devol.
NCRE-2011-2	VENT-2011-1	VIRGINIA PIMBO CHINGO	\$1,494.48	\$58.24
NCRE-2011-2	VENT-2011-2	GERARDO FLORES LOPEZ	\$1,378.00	\$58.24
NCRE-2011-2	VENT-2011-3	GERARDO FLORES LOPEZ	\$1,378.00	\$58.24
NCRE-2011-2	VENT-2011-4	GABRIELA MARANON BURBANO	\$1,350.44	\$58.24
NCRE-2011-1	VENT-2011-5	ANA FLORES PIMBO	\$1,290.24	\$1,290.24

Figura 1D. Listado de Devolución de Ventas

Registro de devoluciones

Al seleccionar la opción **Nuevo** se visualiza el siguiente formulario Figura 2D, donde debemos llenar los campos requeridos para ingresar una nueva devolución.

Figura 2D. Ingreso nueva Devolución

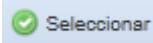
Dentro del formulario Figura 2D al dar click en el botón Buscar  podemos visualizar el Listado de Facturas Figura 3D donde debemos seleccionar la factura de la cual se hará la devolución; para agregar los productos a devolver en el presionamos el botón Añadir  el visualizará el Listado de Productos Vendidos Figura 4D según la factura anteriormente seleccionada; y para eliminar un ítem del detalle de la devolución seleccionamos el ítem y luego presionamos el botón Eliminar .

Además se puede adjuntar archivos presionando el botón .

Al guardar la información se visualizara en el Listado de Devolución de ventas Figura 1D.

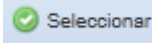
Nº	CVRUC	Cliente	Fecha emision	Total venta	Forma de pago
VENT-2011-4	1801051150	VIRGINIA PIMBO CHINGO	12/11/2011	\$1,350.44	CONTADO
VENT-2011-2	1804192837001	GABRIELA MARANON BURBANO	12/11/2011	\$1,378.00	CREDITO
VENT-2011-1	1804192837	ANA FLORES PIMBO	12/11/2011	\$1,494.48	CONTADO

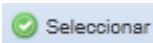
Figura 3D. Listado de Facturas

Para seleccionar un datos del formulario Figura 3D debemos dar doble click sobre el registro deseado o seleccionarlo y luego presionar el botón .


<input type="checkbox"/>	Código	Descripción	Cantidad	V. Unitario	% Descu...	Total	Codigo Series
<input type="checkbox"/>	TCL-001	TECLADO	1	\$39.00	0	\$39.00	
<input type="checkbox"/>	MS-001	MOUSE	1	\$52.00	0	\$52.00	BSER6
<input type="checkbox"/>	CPU-001	CPU	1	\$1,128.96	2	\$1,128.96	
<input type="checkbox"/>	FORMT_001	FORMATEO DE DISCO DURO	1	\$10.40	0	\$10.40	

Figura 4D. Listado de productos vendidos

Para seleccionar un datos del formulario Figura 4D debemos dar doble click sobre el registro deseado o seleccionarlo y luego presionar el botón . Y para seleccionar varios registros marcamos los datos deseados y presionamos el botón



Edición de devoluciones

Para editar un registro del Listado de Devolución de Ventas Figura 1D, se debe seleccionar el ítem y presionar el botón Editar  de la Figura 1D, o dar doble click sobre el ítem seleccionado.

Se podrá editar los siguientes campos:

- Fecha que caduca
- Adjunto
- Observación



Devolución de venta
El formulario permite administrar la información de la tabla Devoluciones

Datos de la factura

Nº factura: VENT-2011-2
Cliente: GABRIELA MARANON BURBANO
Total venta: 1378
A contado \$: 1178 A crédito \$: 200

Datos de la devolución

Nº: 0000001 Valor a favor: 1166.36
Fecha emis: 22/11/2011 Fecha caduca: 22/11/2011
Adjunto: DIAGRAMA_SISAC.pdf
Observación: Modificación

Detalle

Añadir Eliminar


Código	Descripción	Cantidad	% Desc.	V. Unit.	Total	Ser
1	TCL-001 TECLADO	1	0	\$39.00	\$39.00	
2	MS-001 MOUSE	1	0	\$52.00	\$52.00	DDI
3	CPU-001 CPU	1	2	\$1,128.96	\$1,128.96	

Subtotal : 1219.96
Desc 0 % : 0
Iva 12 % : 146.4
TOTAL: 1366.36

Guardar Cancelar Cerrar

Figura 5D. Edición de la devolución

Eliminación de devoluciones

Para eliminar una devolución del listado Figura 1D, debemos seleccionar la devolución a eliminar y presionar el botón Eliminar  del formulario Figura 1D. Se visualizará el siguiente mensaje Figura 6D para confirmar la eliminación del registro.

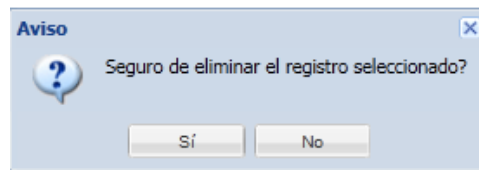


Figura 6D. Mensaje de verificación de Eliminación de la retención



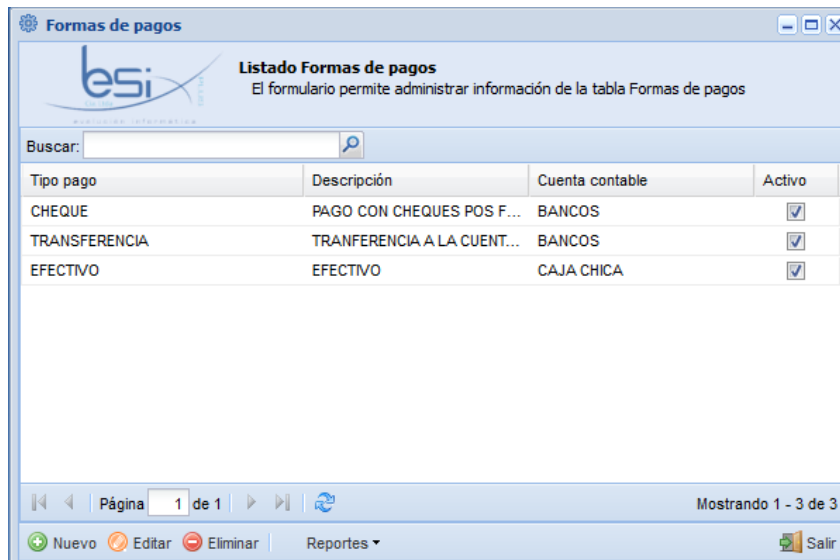
FORMAS DE PAGO

En esta parte se explicará acerca de la administración de la información relacionada formas de pago.

Acceso a la opción de formas de pago

Para ingresar a esta pantalla podemos hacerlo desde el menú Inicio eligiendo la opción Opciones Generales “Formas de pago”.

En el formulario Figura 1E podemos visualizar la información de las formas de pago registradas.



The screenshot shows a web application window titled 'Formas de pagos'. The main heading is 'Listado Formas de pagos' with a subtitle 'El formulario permite administrar información de la tabla Formas de pagos'. There is a search bar labeled 'Buscar:'. Below it is a table with four columns: 'Tipo pago', 'Descripción', 'Cuenta contable', and 'Activo'. The table contains three rows of data. At the bottom of the window, there is a navigation bar with 'Página 1 de 1', 'Mostrando 1 - 3 de 3', and buttons for 'Nuevo', 'Editar', 'Eliminar', 'Reportes', and 'Salir'.

Tipo pago	Descripción	Cuenta contable	Activo
CHEQUE	PAGO CON CHEQUES POS F...	BANCOS	<input checked="" type="checkbox"/>
TRANSFERENCIA	TRANSFERENCIA A LA CUENT...	BANCOS	<input checked="" type="checkbox"/>
EFFECTIVO	EFFECTIVO	CAJA CHICA	<input checked="" type="checkbox"/>

Figura 1E. Listado de Formas de Pago

Registro de formas de pago

Al seleccionar la opción **Nuevo** del formulario Listado de Formas de Pago Figura 1E se visualiza el siguiente formulario Figura 2E, donde debemos llenar los campos requeridos para ingresar una nueva Forma de Pago.

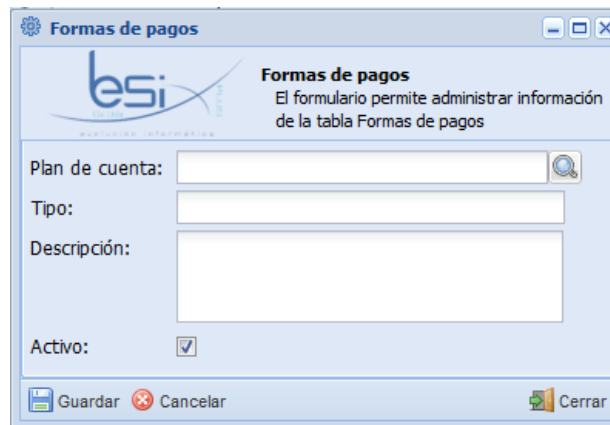




Figura 2E. Ingreso nueva Forma de Pago

Cada forma de pago va relacionada con una cuenta contable para eso debemos presionar el botón  y se visualizará el Listado de Cuentas Contables Figura 3E aquí debemos seleccionar la cuenta contable relacionada con la forma de pago y presionar el botón  Seleccionar .

Al guardar la información se visualizará en el Listado de formas de pago Figura 1E.

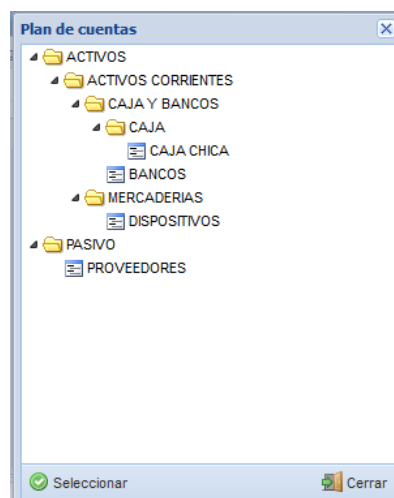

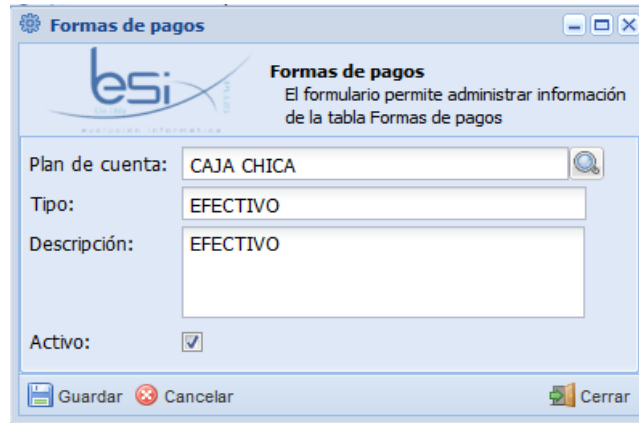


Figura 3E. Listado de Cuentas Contables

Edición de formas de pago

Para editar un registro del Listado de Formas de Pago debemos seleccionar el ítem y presionar el botón Editar  de la Figura 1E, o dar doble click sobre el ítem; se visualizará el formulario Figura 4E donde podremos modificar todos los campos del formulario.



Formas de pagos

El formulario permite administrar información de la tabla Formas de pagos

Plan de cuenta: CAJA CHICA

Tipo: EFECTIVO


Descripción: EFECTIVO

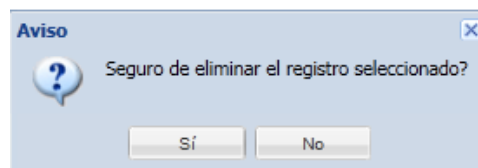
Activo:

Guardar Cancelar Cerrar

Figura 4E. Edición de Forma de Pago.

Eliminación de formas de pago

Para eliminar una Forma de Pago del listado Figura 1E, debemos seleccionar la Forma de Pago a eliminar y presionar el botón Eliminar  del formulario Figura 1E. Se visualizará el siguiente mensaje Figura 5E para confirmar la eliminación del registro.



Aviso

Seguro de eliminar el registro seleccionado?

Sí No

Figura 5E. Mensaje de verificación de Eliminación de la Forma de Pago

ADMINISTRACIÓN DE INVENTARIOS



KARDEX

En esta parte se explicará acerca de la administración de la información relacionada con las tarjetas kardex.

Una tarjeta kardex está formada por la unión de un Producto y una o varias bodegas; la kardex se crea al momento de crear el Producto en el Módulo Compras.


Acceso a la opción de Kardex.

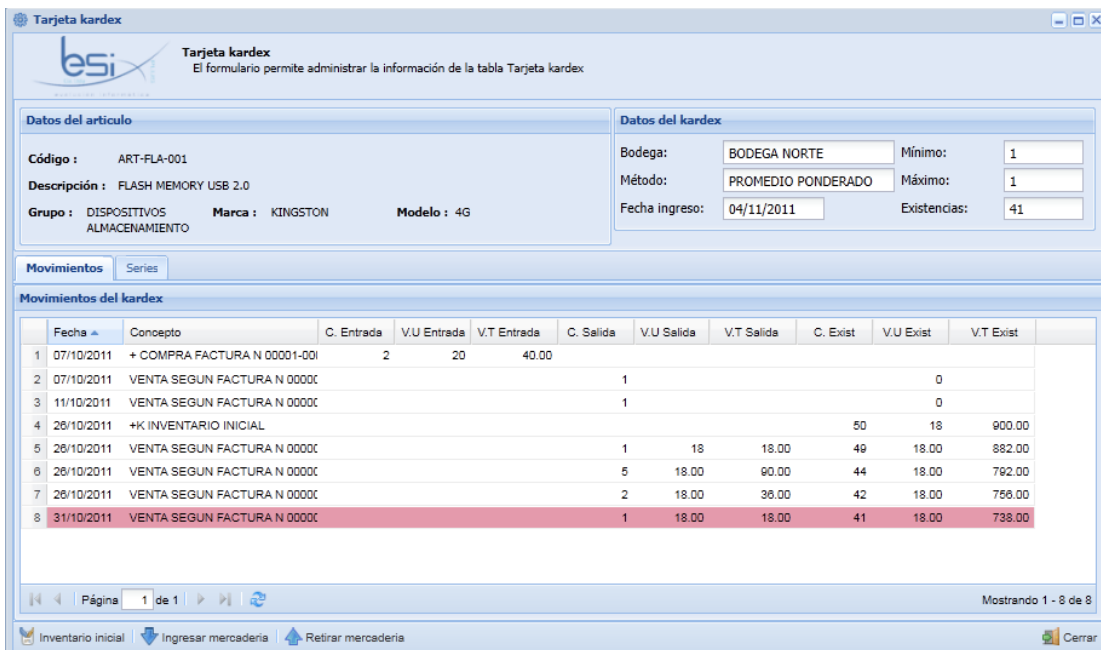
Para ingresar a esta pantalla podemos hacerlo desde el acceso directo en el escritorio “Kardex” o por el menú Inicio eligiendo la opción Gestión de Inventarios “Kardex” se visualizará el formulario Listado de Kardex Figura 1F donde podemos visualizar la información de las Kardex registradas.

Nº kardex	Artículo	Código	Metodo	Mínimo	Máximo	Existen...
000001	TECLADO	TCL-001	PROMEDIO PONDERADO	1	30	36
000002	MOUSE	MS-001	PROMEDIO PONDERADO	1	20	5
000003	CPU	CPU-001	PROMEDIO PONDERADO	1	40	46

Figura 1F. Listado de Kardex

Detallar movimientos de la tarjeta kardex

Para poder detallar los movimientos registrados en la tarjeta Kardex debemos seleccionar el Producto del Listado de Kardex Figura 1F y presionar el botón Detallar  , o dar doble click sobre el ítem seleccionado; se visualizará el formulario de Movimientos del Kardex Figura 2F.



The screenshot shows the 'Tarjeta kardex' application window. It has a header with the 'bsi' logo and the text 'Tarjeta kardex El formulario permite administrar la información de la tabla Tarjeta kardex'. Below the header are two sections: 'Datos del artículo' and 'Datos del kardex'. 'Datos del artículo' contains fields for 'Código', 'Descripción', 'Grupo', 'Marca', and 'Modelo'. 'Datos del kardex' contains fields for 'Bodega', 'Método', 'Fecha ingreso', 'Mínimo', 'Máximo', and 'Existencias'. Below these sections are two tabs: 'Movimientos' (selected) and 'Series'. The 'Movimientos del kardex' section contains a table with the following data:

Fecha	Concepto	C. Entrada	V.U Entrada	V.T Entrada	C. Salida	V.U Salida	V.T Salida	C. Exist	V.U Exist	V.T Exist
1 07/10/2011	+ COMPRA FACTURA N 00001-001	2	20	40.00						
2 07/10/2011	VENTA SEGUN FACTURA N 00000				1				0	
3 11/10/2011	VENTA SEGUN FACTURA N 00000				1				0	
4 26/10/2011	+K INVENTARIO INICIAL							50	18	900.00
5 26/10/2011	VENTA SEGUN FACTURA N 00000				1	18	18.00	49	18.00	882.00
6 26/10/2011	VENTA SEGUN FACTURA N 00000				5	18.00	90.00	44	18.00	792.00
7 26/10/2011	VENTA SEGUN FACTURA N 00000				2	18.00	36.00	42	18.00	756.00
8 31/10/2011	VENTA SEGUN FACTURA N 00000				1	18.00	18.00	41	18.00	738.00

At the bottom of the window, there are navigation buttons: 'Inventario inicial', 'Ingresar mercadería', 'Retirar mercadería', and 'Cerrar'. The status bar shows 'Página 1 de 1' and 'Mostrando 1 - 8 de 8'.

Figura 2F. Listado de Movimientos del Kardex


En la Figura 2F se mostrará todos los movimientos registrados del Producto en la Bodega designada; los datos podrán ser ordenados por todos los campos del listado de forma ascendente o descendente.

Además si el Producto maneja series se podrá revisar las Series del Producto seleccionando la opción Series del tabPanel.

REGISTRO DE MOVIMIENTOS EN LA TARJETA KARDEX

Inventario inicial

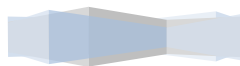
Para registrar un nuevo Inventario Inicial del Producto debemos seleccionar la opción Inventario Inicial del formulario Listado de Movimientos del Kardex Figura 2F, se visualizará el formulario Ingreso nuevo Inventario Inicial Figura 3F, donde debemos llenar los campos requeridos para almacenar un nuevo Inventario inicial.

Si el Producto del cual se va a ingresar el nuevo Inventario Inicial maneja series se deberá ingresar las series y automáticamente la cantidad de series que se ingrese pasará al campo Cantidad; también se puede hacer la búsqueda de series anteriormente registradas presionando el botón Buscar  se visualizará el formulario Listado de series Figura 4F.

Si el producto no maneja series se deberá ingresar directamente en el campo Cantidad la cantidad de productos para el Inventario Inicial.

El campo Valor Total es el resultado del campo Cantidad y el campo Valor Unitario.

Una vez guardar la información se mostrará en el detalle de los movimientos del Producto Figura 2F.



Inventario inicial

Inventario inicial
El formulario permite administrar la información de movimientos de la mercadería

Datos del artículo

Código : ART-TEC-001 Existencias actuales : 9

Descripción : TECLADO TECLADO ESPAÑOL ESTANDAR

Datos del movimiento

Fecha: 04/11/2011

Concepto: INVENTARIO INICIAL

Cantidad: 0

Valor unitario: 1

Valor total: 0

Observación:

Ciente o proveedor:

Cliente

Proveedor

Series

Serie: + Añadir - Eliminar 🔍 Buscar

Serie	Observación
-------	-------------

Guardar Cancelar Cerrar

Figura 3F. Ingreso nuevo Inventario Inicial

Series del artículo -> MS-001

Buscar: 🔍

<input type="checkbox"/>	Serie
<input type="checkbox"/>	GHFDDDD
<input type="checkbox"/>	EFGGHHH
<input type="checkbox"/>	DDDCCC
<input type="checkbox"/>	SXSSSS
<input type="checkbox"/>	DEEEEEE
<input type="checkbox"/>	RRYTFDSDFG
<input type="checkbox"/>	AAAAAXXX
<input type="checkbox"/>	QQQQXXX
<input type="checkbox"/>	WWWWEEE
<input type="checkbox"/>	ASDFGH

« ‹ Página 1 de 1 › » ↻ Mostrando 1 - 12 de 12




✓ Seleccionar Cerrar

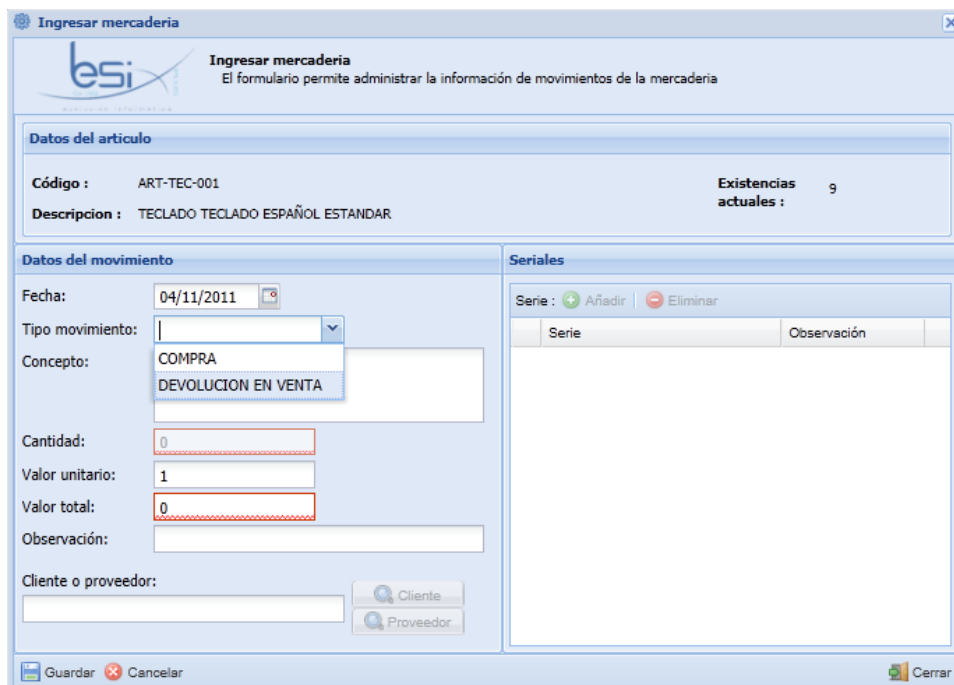
Figura 4F. Listado de Series del Producto

Ingreso de mercadería

Para registrar un nuevo ingreso debemos seleccionar la opción Ingresar Mercadería del formulario Listado de Movimientos del Kardex Figura 2F,

Esta opción permite registrar un movimiento a la Kardex de forma manual, mediante una Compra Figura 5F o una Devolución Figura 6F.

- Si el ingreso es por medio de una Compra Figura 5F simplemente se deberá llenar los campos requeridos y presionar el botón Guardar 
- Si el ingreso es por medio de una Devolución Figura 6F se deberá buscar la Venta de la cual se va a realizar la Devolución presionando el botón  Buscar venta se visualizará el formulario Ventas registradas en Kardex Figura 7F. Donde debemos seleccionar la venta a Devolver y automáticamente se pasaran los datos a los campos de la Figura 6F una vez verificado los datos presionamos el botón Guardar 



Ingresar mercadería

besixplus **Ingresar mercadería**
El formulario permite administrar la información de movimientos de la mercadería

Datos del artículo

Código : ART-TEC-001 Existencias actuales : 9
Descripción : TECLADO TECLADO ESPAÑOL ESTANDAR

Datos del movimiento

Fecha: 04/11/2011
Tipo movimiento:
Concepto: COMPRA
DEVOLUCION EN VENTA
Cantidad: 0
Valor unitario: 1
Valor total: 0
Observación:
Cliente o proveedor:
Cliente
Proveedor

Seriales

Serie : Añadir Eliminar

Serie	Observación
-------	-------------

Guardar Cancelar Cerrar

Figura 5F. Ingreso de mercadería por Compra

Ingresar mercadería

bsi Ingresar mercadería
El formulario permite administrar la información de movimientos de la mercadería

Datos del artículo

Código : ART-TEC-001 Existencias actuales : 9

Descripción : TECLADO TECLADO ESPAÑOL ESTANDAR

Datos del movimiento

Fecha: 04/11/2011

Tipo movimiento: DEVOLUCION EN VENT/ Buscar venta

Concepto: DEVOLUCIONVENTA SEGUN FACTURA N 000005

Cantidad: 1

Valor unitario: 1

Valor total: 1

Observación:

Cliente o proveedor: BCLIE Cliente Proveedor

Guardar Cancelar Cerrar

Serie

Serie : Añadir Eliminar

Serie	Observación
1 WERYYY	

Figura 6F. Ingreso de mercadería por Devolución

Buscar

Buscar:

Concepto	Cant. Entrada	V.U Entrada	V.T Entrada	Cant. Salida	V.U Salida
VENTA SEGUN FACTURA N 000004				1	8.00
VENTA SEGUN FACTURA N 000002				1	8.00
VENTA SEGUN FACTURA N 000001				3	8

Mostrando 1 - 3 de 3




Seleccionar Cerrar

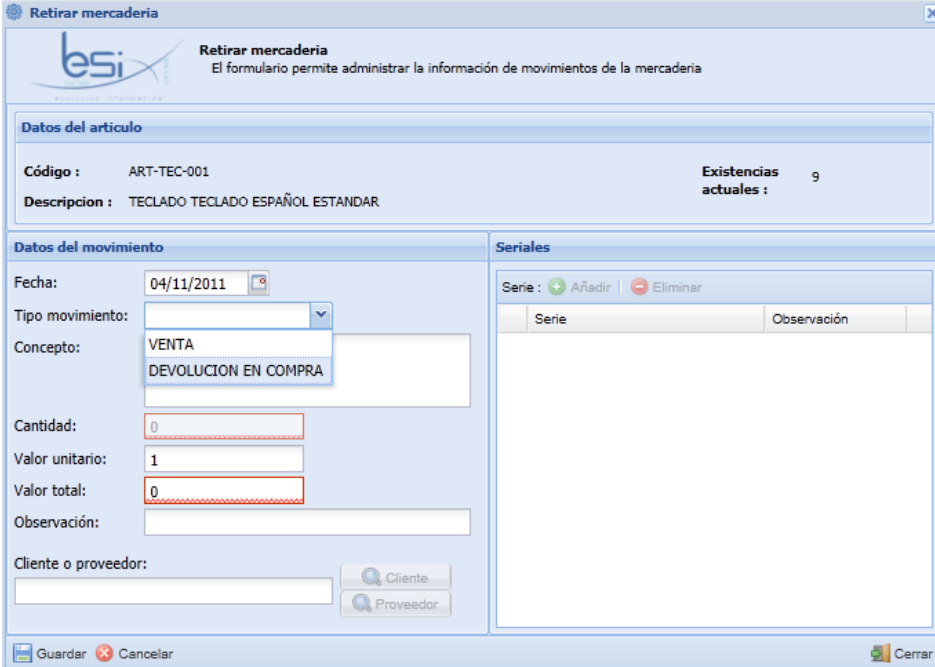
Figura 7F. Ventas registradas en Kardex

Retiro de mercadería

Para registrar una nueva salida debemos seleccionar la opción Retirar Mercadería del formulario Listado de Movimientos del Kardex Figura 2F,

Esta opción permite registrar un movimiento a la Kardex de forma manual, mediante una Venta Figura 8F o una Devolución Figura 9F.

- Si el retiro es por medio de una Venta Figura 8F simplemente se deberá llenar los campos requeridos y presionar el botón Guardar 
- Si el ingreso es por medio de una Devolución Figura 9F se deberá buscar la Compra de la cual se va a realizar la Devolución presionando el botón  Buscar compra se visualizará el formulario Compras registradas en Kardex Figura 10F. Donde debemos seleccionar la compra a Devolver y automáticamente se pasaran los datos a los campos de la Figura 9F una vez verificado los datos presionamos el botón Guardar 



Serie	Observación
-------	-------------

Figura 8F. Salida de mercadería por Venta

Retirar mercadería

El formulario permite administrar la información de movimientos de la mercadería

Datos del artículo

Código : ART-TEC-001 Existencias actuales : 9
 Descripción : TECLADO TECLADO ESPAÑOL ESTANDAR

Datos del movimiento

Fecha: 04/11/2011
 Tipo movimiento: DEVOLUCION EN COMP Buscar compra
 Concepto: DEVOLUCION+K. COMPRA SEGUN FACTURA N 000234566
 Cantidad: 3
 Valor unitario: 1
 Valor total: 3
 Observación:
 Cliente o proveedor: BCPRO1 Cliente Proveedor

Serieles

Serie : + Añadir - Eliminar

Serie	Observación
1 EDRFVC345	
2 WEERTY3456	
3 QWERDC345	

Guardar Cancelar Cerrar

Figura 9F. Salida de mercadería por Devolución

Buscar

Buscar:

Concepto	Cant. Entrada	V.U Entrada	V.T Entrada	Cant. Salida	V.U Salida	V.T
+K COMPRA SEGUN FACTUR...	40	12	480.00			
+K COMPRA SEGUN FACTUR...	40	13	520.00			
+K COMPRA SEGUN FACTUR...	50	12	600.00			

Mostrando 1 - 3 de 3

Seleccionar Cerrar

Figura 10F. Compras registradas en Kardex

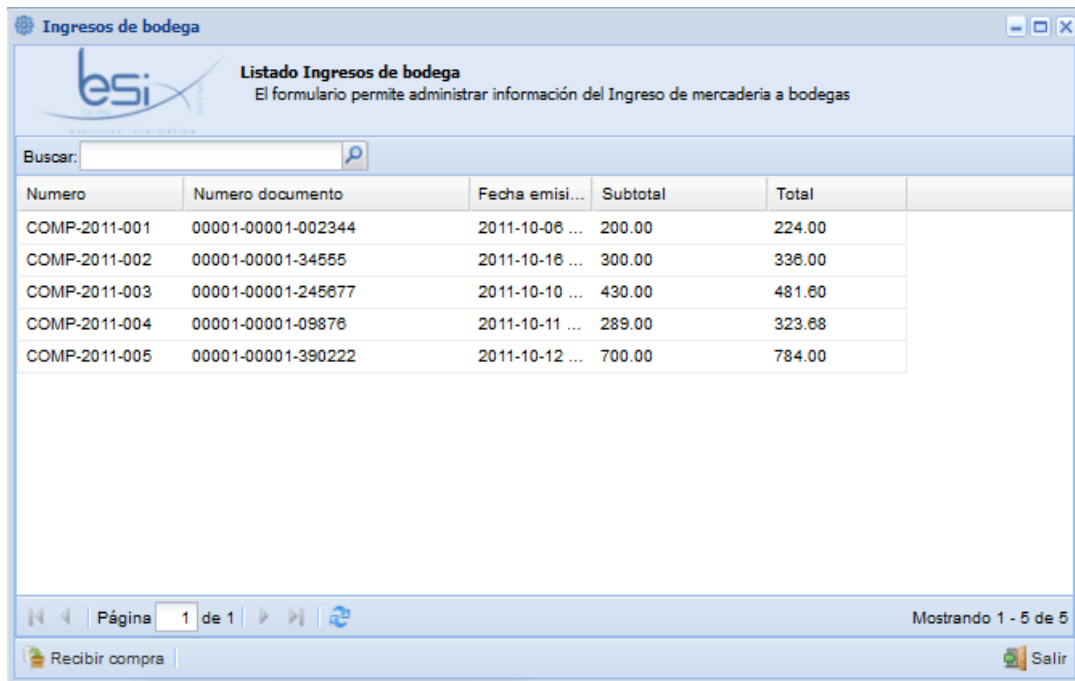
INGRESOS DE BODEGA

En esta parte se explicará acerca de la administración de la información relacionada con los Ingresos de Bodega.

Para realizar un Ingreso de Bodega se debe tener compras registradas en el Módulo de Compras, las compras deben estar en estado Aprobado para poder ser ingresadas.

Acceso a la opción de ingresos de bodega.

Para ingresar a esta pantalla podemos hacerlo desde el menú Inicio eligiendo la opción Gestión de Inventarios “Ingresos de Bodega”, se visualizará el Listado de Compras Aprobadas Figura 1G

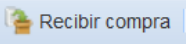


The screenshot shows a web application window titled 'Ingresos de bodega'. The main heading is 'Listado Ingresos de bodega' with a subtitle 'El formulario permite administrar información del Ingreso de mercadería a bodegas'. Below the heading is a search bar labeled 'Buscar:'. The main content is a table with the following data:

Numero	Numero documento	Fecha emisi...	Subtotal	Total
COMP-2011-001	00001-00001-002344	2011-10-06 ...	200.00	224.00
COMP-2011-002	00001-00001-34555	2011-10-16 ...	300.00	336.00
COMP-2011-003	00001-00001-245677	2011-10-10 ...	430.00	481.60
COMP-2011-004	00001-00001-09876	2011-10-11 ...	289.00	323.68
COMP-2011-005	00001-00001-390222	2011-10-12 ...	700.00	784.00

At the bottom of the window, there is a pagination control showing 'Página 1 de 1' and 'Mostrando 1 - 5 de 5'. There are also buttons for 'Recibir compra' and 'Salir'.

Figura 1G. Listado de Compras Aprobadas

Para hacer el ingreso a Bodega de una Compra debemos seleccionar un registro del Listado de Compras Aprobadas de la Figura 1G y presionar el botón  o dar doble click en ítem seleccionado; se visualizará el formulario Nuevo Ingreso a Bodega Figura 2G.

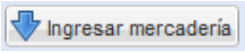


	Código	Descripción	Cantidad	V. unit...	V. total
1	TCL-001	TECLADO	3	\$12.00	\$24.00
2	CPU-001	CPU	4	\$34.00	\$24.00
3	MS-001	MOUSE	4	\$34.00	\$24.00

Figura 2G. Nuevo Ingreso a Bodega

En el formulario Nuevo Ingreso a Bodega Figura 2G debemos seleccionar el producto a Ingresar, los datos del ítem seleccionado se visualizarán en el panel Artículo Figura 3G, aquí se debe seleccionar la Bodega donde se almacenará el producto y presionar

el botón .

Si el Producto a Ingresar maneja series se deberán ingresar las series del artículo en Ingreso de Series Figura 4G; la cantidad de series ingresadas debe coincidir con la cantidad de Producto adquirido una vez ingresadas las series y seleccionada la bodega presionamos el botón .

Artículo

Artículo:

Cantidad:

V. unit:

V. total:

Bodega :

Observación:

Figura 3G. Panel Artículo

Artículo

Artículo:

Cantidad:

V. unit:

V. total:

Bodega :

Observación:

Series

Serie :

	Serie	Observación
1	XCCX	
2	CXXC	
3	ASDA	
4	ASDAS	

Figura 4G. Ingreso de Series

Toda la información registrada en el Ingreso a Bodega Figura 2G se insertará como un movimiento en la tarjeta Kardex del Producto ingresado.

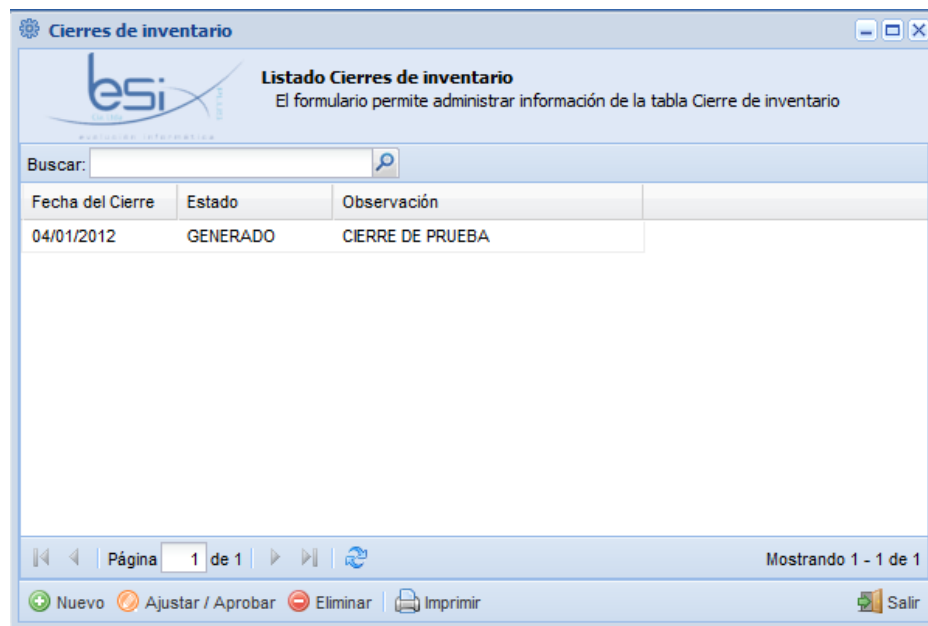
TOMA Y AJUSTE DE INVENTARIO

En esta parte se explicará acerca de la administración de la información relacionada con la toma y ajuste de inventario.

Para realizar una toma de inventario no debe hacer tomas de inventarios en estado **Generado**.

Acceso a la opción de toma y ajuste de inventario.

Para ingresar a esta pantalla podemos hacerlo desde el menú Inicio eligiendo la opción Gestión de Inventarios “Cierre de inventario”, se visualizará el Listado de Cierres de Inventarios Figura 1H.

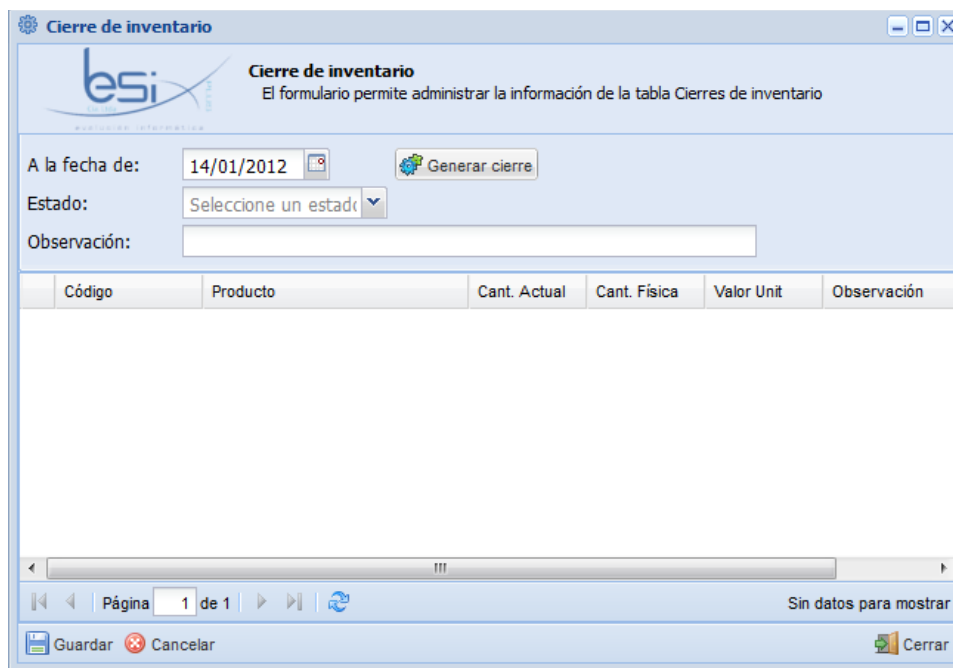


Fecha del Cierre	Estado	Observación
04/01/2012	GENERADO	CIERRE DE PRUEBA

Figura 1H. Listado de Cierres de Inventario

Registro de toma de inventario

Al seleccionar la opción **Nuevo** del formulario Listado de Cierres de Inventario Figura 1H se visualiza el siguiente formulario Figura 2H, donde debemos llenar los campos requeridos para ingresar una nueva Toma de Inventario.



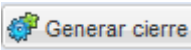
The screenshot shows a web application window titled "Cierre de inventario". The window contains a header with the "bsi" logo and the text "Cierre de inventario" and "El formulario permite administrar la información de la tabla Cierres de inventario". Below the header, there are several input fields and buttons:

- A date field labeled "A la fecha de:" with the value "14/01/2012" and a calendar icon.
- A button labeled "Generar cierre" with a gear icon.
- A dropdown menu labeled "Estado:" with the text "Seleccione un estado".
- A text input field labeled "Observación:".

Below these fields is a table with the following columns: "Código", "Producto", "Cant. Actual", "Cant. Física", "Valor Unit", and "Observación". The table is currently empty.

At the bottom of the window, there is a navigation bar with a "Página 1 de 1" indicator, a "Sin datos para mostrar" message, and buttons for "Guardar", "Cancelar", and "Cerrar".

Figura 2H. Ingreso nueva Toma de Inventario

Las tomas de inventario se hacen a la fecha, ya establecida la fecha del cierre presionamos el botón  el cual visualizará la información de los artículos y las cantidades existentes en cada bodega Figura 3H, luego seleccionamos el estado con el cual se guardará la toma de inventario pudiendo escoger entre: GENERADO, APROBADO O RECHAZADO.

Cierre de inventario
El formulario permite administrar la información de la tabla Cierres de inventario

A la fecha de: 14/01/2012

Estado: GENERADO

Observación:

	Código	Producto	Cant. Actual	Cant. Física	Valor Unit	Observación
Bodega: BODEGA1						
1	ART-DIS-2011-1	TECLADO	25	0	0	
2	ART-DIS-2011-2	MOUSE	5	0	0	
3	ART-DIS-2011-3	SCANNER	7	0	0	
4	ART-DIS-2011-4	MONITOR	1	0	0	
5	ART-DIS-2011-5	IMPRESORA	2	0	0	
Bodega: BODEGA2						
6	ART-DIS-2011-6	DISCO DURO	10	0	0	


Página 1 de 1

Mostrando 1 - 7 de 7


Figura 3H. Información obtenida al Generar Cierre.

Al guardar la información se visualizará en el Listado de Cierres de inventario Figura 1H.

Ajuste y aprobación de la toma de inventario

Para AJUSTAR o APROBAR una toma de inventario, se debe seleccionar un registro del Listado de Cierres de Inventario el cual debe estar en estado GENERADO y presionar el botón Ajustar/Aprobar  de la Figura 1H, o dar doble click sobre el ítem; se visualizará el formulario Figura 3H donde podremos ingresar las CANTIDADES FÍSICAS y el VALOR UNITARIO de cada Artículo, además seleccionamos el estado APROBADO O RECHAZADO, una vez almacenada la información se actualizarán los datos en la TARJETA KARDEX.

Eliminación de una toma de inventario

Para eliminar una TOMA DE INVENTARIO del listado Figura 1H, debemos seleccionar la Toma de Inventario a eliminar y presionar el botón Eliminar  del formulario Figura 1H. Se visualizará el siguiente mensaje Figura 4H para confirmar la eliminación del registro.

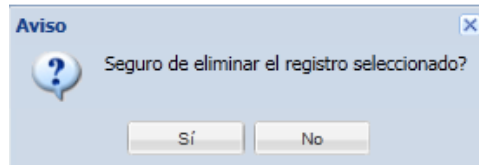


Figura 4H. Mensaje de verificación de Eliminación de la Toma de Inventario

