



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA
COMPUTACIÓN**

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO
BASADO EN TECNOLOGÍA RFID (RADIO FRECUENCIA E
IDENTIFICACIÓN) Y GEOLOCALIZACIÓN PARA EL
CONTROL DEL STOCK Y GESTIÓN DEL INVENTARIO POR
LOTES, PARA LA EMPRESA ZEUZ SISTEMAS**

**AUTORES: GUALOTO SUÁREZ FRANZ MICHAEL
ORTEGA ARGÜELLO RUBÉN GERARDO**

DIRECTOR: ING. ÑACATO CAIZA GERMAN

**SANGOLQUÍ
JULIO 2016**

CERTIFICACIÓN



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

CERTIFICACIÓN

Certifico que el trabajo de titulación, "IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO BASADO EN TECNOLOGÍA RFID (RADIO FRECUENCIA E IDENTIFICACIÓN) Y GEOLOCALIZACIÓN PARA EL CONTROL DEL STOCK Y GESTIÓN DEL INVENTARIO POR LOTES, PARA LA EMPRESA ZEUS SISTEMAS", ha sido revisado en su totalidad y analizado por el software antiplagio el mismo cumple con los requisitos teóricos y científicos técnicos metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas – ESPE por lo tanto me permito acreditarlo y autorizarlo a los Señores: GUALOTO SUÁREZ FRANZ MICHAEL Y ORTEGA ARGÜELLO RUBÉN GERARDO, para que lo sustente públicamente.

Sangolquí, 14 de julio de 2016



Jrte. Germán Nacato
DIRECTOR

AUTORÍA DE RESPONSABILIDAD

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA
AUTORÍA DE RESPONSABILIDAD

Nosotros **FRANZ MICHAEL GUALOTO SUÁREZ**, con cedula de identidad 172273733-3 y **RUBÉN GERARDO ORTEGA ARGÜELLO**, con cedula de identidad 171695229-4 declaramos que este trabajo de titulación **"IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO BASADO EN TECNOLOGÍA RFID (RADIO FRECUENCIA E IDENTIFICACIÓN) Y GEOLOCALIZACIÓN PARA EL CONTROL DEL STOCK Y GESTIÓN DEL INVENTARIO POR LOTES, PARA LA EMPRESA ZEUS SISTEMAS"**, ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaramos que este trabajo es de nuestra autoría, en virtud de ello declaramos ser responsables del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 14 de julio de 2016

Franz Michael Gualoto Suárez

Ci: 172273733-3

Rubén Gerardo Ortega Argüello

Ci: 171695229-4

AUTORIZACIÓN

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA**

AUTORIZACIÓN

Nosotros **FRANZ MICHAEL GUALOTO SUÁREZ** y **RUBÉN GERARDO ORTEGA ARGÜELLO**, Autorizamos a la Universidad de las Fuerzas Armadas – ESPE, publicar en la Biblioteca virtual de la institución el presente trabajo de titulación **“IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO BASADO EN TECNOLOGÍA RFID (RADIO FRECUENCIA E IDENTIFICACIÓN) Y GEOLOCALIZACIÓN PARA EL CONTROL DEL STOCK Y GESTIÓN DEL INVENTARIO POR LOTES, PARA LA EMPRESA ZEUS SISTEMAS”**, cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Sargolquí, 14 de Julio del 2016

Franz Michael Gualoto Suárez

Ci: 172273733-3

Rubén Gerardo Ortega Argüello

Ci: 171695229-4

DEDICATORIA

A mis padres

Por ser un gran apoyo en mi desempeño profesional, y por brindarme la mejor educación de la vida la sencillez y la humildad.

A mi hermano

Aquel gran ser humano que siempre ha estado en las más grandes batallas que se ha presentado.

A mi novia

La persona que en este tiempo siempre me ha impulsado a grandes cosas.

A mi Angelito

Quien en su momento supo darme gran felicidad, pero fue el mejor regalo que la vida me haya dado.

AGRADECIMIENTO

Con todo mi corazón agradezco a Dios por darme la fortaleza de seguir adelante en cada momento de la vida, a mis padres Vicente y Cristina que siempre me han apoyado, sobre todo los valores que siempre me han inculcado ya que son un pilar muy importante en mi vida.

Franz Gualoto

DEDICATORIA

El presente trabajo dedico a mis padres Honorio Ortega y Pilar Argüello por brindarme todo su apoyo incondicional, y sus sabios consejos, para que cumpla con uno de mis objetivos propuestos a lo largo de mi vida. A mi esposa Jenny Chuquitarco y mis dos hijos Ariel y Gerard que son el motor fundamental de mi existencia, ya que gracias a ellos tengo la fuerza para seguir luchando en el día a día. A mis hermanos, Héctor, Fernando y Byron que siempre están prestos a darme su apoyo sin escatimar ningún esfuerzo.

Rubén Ortega.

AGRADECIMIENTO

Agradezco a Dios por darme la vida y la oportunidad de superarme siendo un hombre de bien. A mis padres ya que ellos me dieron la herencia más preciada que el hombre puede tener “La Educación”, A la Universidad de las Fuerzas Armadas y a todos los docentes por darme una segunda oportunidad para culminar mis estudios e inculcarme todo el conocimiento y valores que me permitieron formarme y cumplir con ésta meta trazada en mi vida. A mis amigos incondicionales Franz Gualoto, Juan Carlos Chiluisa, y Marisol Oña ya que ellos me apoyaron desde el inicio hasta el último día de mi vida universitaria.

Rubén Ortega.

ÍNDICE

CERTIFICACIÓN	i
AUTORÍA DE RESPONSABILIDAD	ii
AUTORIZACIÓN	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
RESUMEN	xiv
ABSTRACT	xv
CAPÍTULO I	1
INTRODUCCIÓN	1
1.1 ANTECEDENTES	1
1.2 PROBLEMÁTICA	2
1.3 JUSTIFICACIÓN.....	3
1.4 OBJETIVOS.....	4
1.5 ALCANCE	4
1.6 FACTIBILIDAD	6
1.6.1 Factibilidad técnica	6
1.6.2 Requerimientos de hardware.....	7
1.6.3 Requerimientos de software.....	7
1.6.4 Factibilidad Económica.....	7
1.6.5 Factibilidad Operativa.....	9
1.6.6 Factibilidad Operacional	9
1.6.7 Factibilidad legal	10
CAPÍTULO II	11
MARCO CONCEPTUAL	11
2.1 KARDEX	11
2.2 IDENTIFICACIÓN AUTOMÁTICA POR RADIO FRECUENCIA	13

2.2.1	Aplicaciones Radio Frecuencia.....	13
2.2.2	Ventajas de Radio Frecuencia.....	14
2.2.3	Características	15
MARCO TEÓRICO		16
2.3	RFID (RADIO FRECUENCIA E IDENTIFICACIÓN).....	16
2.3.1	COMPONENTES.	17
2.3.2	TIPOS DE ETIQUETAS	18
2.3.3	FUNCIONAMIENTO	20
2.3.4	CONECTIVIDAD	20
2.3.5	ESTÁNDARES.....	21
2.3.6	MIDDLEWARE	22
2.3.7	FRECUENCIAS	25
2.3.8	VENTAJAS	26
2.4	GEO LOCALIZACIÓN Y RFID	26
2.5	CARACTERÍSTICAS DEL RFID VS CÓDIGO DE BARRAS.....	27
2.6	EMPRESA ZEUZ SISTEMAS.....	28
2.6.1	MISIÓN	28
2.6.2	VISIÓN	28
2.7	METODOLOGÍA DE DESARROLLO DE SISTEMAS XP (EXTREME PROGRAMMING).	28
2.7.1	PRÁCTICAS BÁSICAS DE EXTREME PROGRAMMING.....	29
2.7.2	FASES.....	30
2.8	HERRAMIENTAS DE DESARROLLO.....	35
2.9	Herramienta a utilizar	38
CAPÍTULO III.....		39
ANÁLISIS Y DISEÑO DEL SISTEMA		39
3.1	ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE	39
3.1.1	INTRODUCCIÓN	39
3.1.2	Propósito	39
3.1.3	Alcance.....	39
3.1.4	Especificación de requisitos.....	40
3.1.5	Definición de acrónimos	40
3.1.6	Referencias.....	40
3.1.7	Resumen.....	40
3.2	Historias de usuario	41

3.2.1	Concepto	41
3.2.2	Desarrollo	41
3.2.3	Historias de Usuario	41
3.2.4	Definición de prioridades	54
3.2.5	Definición de iteraciones.....	54
3.3	Diagrama entidad relación.....	56
3.4	Diagrama de clases	58
3.5	Diagrama de Despliegue	58
3.6	Diagrama de componentes	59
CODIFICACIÓN Y PRUEBAS.....		60
4.1	Introducción.....	60
4.1.1	Configuración del servidor de aplicaciones GlassFish Server.....	61
4.1.2	Persistencia.....	61
4.1.3	Desarrollo de la aplicación web	67
4.1.4	Anotaciones.....	67
4.1.5	Anotación en el modelo	67
4.1.6	Anotaciones en la vista.....	71
4.1.7	Anotaciones en el control.....	72
CAPÍTULO V.....		74
RESULTADOS OBTENIDOS Y PRUEBAS		74
5.1	Introducción.....	74
5.1.1	Pruebas de aceptación	74
5.1.2	Matriz de Pruebas de aceptación.....	75
CAPÍTULO VI		80
CONCLUSIONES Y RECOMENDACIONES.....		80
6.1	Introducción.....	80
6.1.1	Conclusiones	80
6.1.2	Recomendaciones.....	81
GLOSARIO DE TÉRMINOS.....		82
REFERENCIAS BIBLIOGRAFICAS.....		83

ÍNDICE DE TABLAS.

Tabla 1 Descripción de Hardware	8
Tabla 2 Descripción de Software	8
Tabla 3 Descripción de Recursos Humanos.....	9
Tabla 4 Descripción de Suministros de Oficina.....	9
Tabla 5 Tecnología RFID vs Código de Barras	27
Tabla 6 Lenguajes de Programación – cuadro comparativo	37
Tabla 7 Historia de Usuario 1	42
Tabla 8 Historia de Usuario 2	42
Tabla 9 Historia de Usuario 3	43
Tabla 10 Historia de Usuario 4	44
Tabla 11 Historia de Usuario 5	44
Tabla 12 Historia de Usuario 6	45
Tabla 13 Historia de Usuario 7	46
Tabla 14 Historia de Usuario 8	46
Tabla 15 Historia de Usuario 9	47
Tabla 16 Historia de Usuario 10	48
Tabla 17 Historia de Usuario 11	48
Tabla 18 Historia de Usuario 12	49
Tabla 19 Historia de Usuario 13	50
Tabla 20 Historia de Usuario 14	50
Tabla 21 Historia de Usuario 15	51
Tabla 22 Historia de Usuario 16	52
Tabla 23 Historia de Usuario 17	53
Tabla 24 Historia de Usuario 18	53
Tabla 25 Prueba de aceptación 1	75
Tabla 26 Prueba de aceptación 2.....	76
Tabla 27 Prueba de aceptación 3.....	76
Tabla 28 Prueba de aceptación 4.....	77

Tabla 29 Prueba de aceptación 5.....	77
Tabla 30 Prueba de aceptación 6.....	78
Tabla 31 Prueba de aceptación 7.....	78
Tabla 32 Prueba de aceptación 8.....	79

ÍNDICE DE FIGURAS

Figura 1. Análisis comparativo entre código de barras y Rfid.....	15
Figura 2. Funcionamiento Sistema RFID	16
Figura 3. Diseño de dos tipos de etiquetas.....	17
Figura 4. Composición de una tarjeta	18
Figura 5. Funcionamiento de un sistema RFID.	20
Figura 6. RFID middleware	24
Figura 7. Fases de la metodología XP.....	30
Figura 8. Esquema de las fases de XP.	34
Figura 9. Cuadro comparativo de base de datos MySql y SqlServer.....	36
Figura 10. Diagrama Entidad Relación.....	57
Figura 11. Diagrama de Clases	58
Figura 12. Diagrama de Despliegue.....	59
Figura 13. Diagrama de Despliegue.....	59
Figura 14. Configuración de la persistencia “PERSISTENCE.XML”	61
Figura 15. Unidad de persistencia.....	62
Figura 16. Iniciar Servicio GlassFish.....	63
Figura 17. Administrador del GlassFish	63
Figura 18. Menú creación Pool de Conexiones	64
Figura 19. Ingreso parámetros del Pool de conexiones.	64
Figura 20. Ingreso de parámetros de conexión a la base de datos.	64

RESUMEN

El proyecto presenta el análisis, desarrollo e implementación de una solución software orientada a la web para el control del stock, gestión del inventario por lotes y geolocalización, que en combinación con Tecnología de RFID (radio Frecuencia) permite llevar un control optimizado del inventario y stock, además tiene la capacidad de reconstruir el historial de la utilización o la localización exacta de un artículo o producto. Para el desarrollo del software se utilizó la metodología XP (Extreme Programming) con el objetivo de cubrir todos los requerimientos y cambios sobre los mismos de manera eficiente y eficaz durante la realización de todo el proyecto. El sistema permite grabar y leer datos (Id, nombre, descripción, localización, etc.) en los tags de radio frecuencia a través de dispositivos lector/grabador de Rfid, estos tags adhesivos son adheridos a cada artículo o producto y posteriormente el sistema recolecta la información a través de antenas o recolectores Rfid de manera simultánea y sin necesidad de línea de vista entre emisor y receptor ya que se comunican a través de ondas de radio, ésta información es procesada permitiendo llevar un control automático del stock. Cabe recalcar que el sistema al tener integrado los dispositivos de RFID y conexión a internet lo hace muy robusto y flexible durante la realización de un inventario ya que posibilita la geolocalización y trazabilidad del mismo. El producto de las pruebas realizadas tanto en el software como en los dispositivos Rfid al momento de hacer un inventario en la empresa “Zeuz System” fue positivas teniendo como resultado optimización de tiempo y recursos, garantizando que el inventario y el control de stock este actualizado y en tiempo real permitiéndole a la empresa tomar óptimas decisiones al momento de comprar a los proveedores y facilitando el movimiento oportuno de los artículos necesarios entre bodegas.

Palabras Clave:

- **RFID**
- **READER**
- **CARDS**
- **INVENTORY SYSTEM**

ABSTRACT

The project presents the analysis, development and implementation of a software solution oriented web for stock control, inventory management and batch geolocation, which in combination with RFID technology (radio frequency) can bring an optimized inventory control and stock also has the ability to reconstruct the history of the use or the exact location of an ítem or product. Software development for the XP (Extreme Programming) methodology was used in order to cover all the requirements and changes on them efficiently and effectively for the realization of the entire project way. The system can record and read data (ID, name, description, location, etc.) in the tag radio frequency through reader / writer RFID devices, these sticky tags are attached to each ítem or product and then the system collects information through RFID antennas or collectors simultaneously without line of sight between transmitter and receiver because they communicate through radio waves, allowing this information is processed out automatic stock control. It should be noted that the system to be integrated RFID devices and internet makes it very robust and flexible during an inventory as possible geolocation and traceability. The result of tests carried out on both the software and RFID devices when making an inventory in the company "Zeuz System" was positive resulting optimization of time and resources, ensuring that inventory and stock control this updated and in real time allowing the company to make optimal decisions when buying to suppliers and facilitating the timely movement of needed ítems between warehouses.

KeyWords:

- **RFID,**
- **READER**
- **CARDS**
- **INVENTORY SYSTEM**

CAPÍTULO I

INTRODUCCIÓN

1.1 ANTECEDENTES

A lo largo de los años han surgido tecnologías de auto identificación como la tecnología RFID que fue desarrollada en el siglo XX para generar soluciones a distintas aplicaciones que puedan brindar ayuda y aportar beneficios positivos, entrelazada a distintas tecnologías de comunicación como: teléfonos móviles, redes inalámbricas, comunicaciones por satélite, GPS, etc. (O'Connor, 2001)

La existencia e implementación de aplicaciones basadas en RFID se debe al desarrollo progresivo de tres áreas tecnológicas principales: (Javier I. Portillo García, 2010)

- Electrónica de radiofrecuencia. Necesaria para el desarrollo de las antenas y los sistemas de radiofrecuencia presentes en las etiquetas e interrogadores RFID.
- Tecnologías de la información. En su vertiente de computación (en el lector, en la propia etiqueta y en el sistema de información asociado) y en las comunicaciones para el envío de información entre etiqueta y lector, y entre lector y sistema de información asociado).
- Tecnología de materiales. Necesaria para disminuir los costos de las etiquetas. Teniendo en cuenta las aplicaciones y las características de funcionamiento en sus principales elementos los cuales se describe a continuación:
 - Etiquetas (tags o transpondedores): almacenan la información.
 - Lectores (interrogadores): leen la información de las etiquetas, proporcionándoles energía para que éstas puedan transmitir los datos almacenados en sus microchip internos.
 - Programadores: escriben información en las etiquetas.
 - Middleware: gestiona el intercambio de información.

- Sistema de información: procesa la información del middleware y toma las decisiones oportunas.

Con la evolución de la tecnología y la creación de los semiconductores las etiquetas RFID abrieron paso a ser un producto de consumo masivo gracias a su bajo costo, a su vez se subdividía en 3 categorías o tipos de funcionamiento: (Toro, 2008)

- RFID pasivo cuando no tenían una fuente de alimentación propia
- RFID semi-pasivo cuando utilizaban una pequeña batería asociada
- RFID activo cuando tenían su propia fuente de alimentación.

1.2 PROBLEMÁTICA

La tecnología de código de barras tiene una gran relevancia ya que hace más de 25 años se ha usado en la identificación de objetos, sin embargo hoy en día cuenta con algunas limitaciones para operar a su máximo rendimiento, es por ésta circunstancia que se han optado por otras formas de identificación de objetos para poder satisfacer necesidades y aplicaciones más complejas que pueden brindar la tecnología RFID la cual es capaz de cubrir dichas limitantes que el código de barras las tiene.

Actualmente la empresa Zeuz Sistemas no cuenta con un sistema que permita la supervisión y control de stock de artículos en bodega, además tiene sus datos digitales manejados en hojas dinámicas Excel, lo que ocasiona el bajo rendimiento y la toma ineficaz de decisiones en la empresa.

Existen diversos problemas al momento de controlar la salida y búsqueda de productos y se mencionan a continuación:

- Retraso en el registro de productos ya que los datos a procesar cuentan con una gran cantidad de información.

- Vulnerabilidad en la integridad de los datos en los archivos de control de bienes ya que al momento de la digitación de los registros en ocasiones se duplica, modifica o se registra erróneamente la información.

1.3 JUSTIFICACIÓN

En vista de que se manipulan los productos de forma continua y estos contienen información susceptible, se ve la necesidad de tener un control real y preciso sobre ellos, razón por la cual se implementará una solución para llevar un inventario digital, donde se pueda tener un óptimo control sobre los bienes y artículos, y sobre todo saber qué persona requirió un determinado producto, la ubicación del mismo y en qué momento.

Es imprescindible realizar un inventario periódico, para constatar la existencia física de los bienes con la información almacenada en el sistema, por tal razón se utilizará la tecnología de radio frecuencia ya que ayudará a hacer el inventariado por lotes de los productos, optimizando tiempo y recursos.

La tecnología RFID (Radio Frecuencia) tiene una amplia área de operación es decir funciona en un radio determinado hasta donde las ondas de radio tengan alcance y no requiere una visión directa con los bienes, para su identificación mientras estén dentro de su área de cobertura.

La seguridad y control de bienes es muy importante dentro de la Empresa por tal razón se debe aplicar mecanismos que permitan identificar, controlar y localizar cada uno de los productos. Mediante la tecnología RFID (Radio Frecuencia) y se puede tener mayor precisión a la hora de realizar este proceso.

1.4 OBJETIVOS

a. Objetivo General

- Implementar un sistema informático para el control y gestión del inventario por lotes de los artículos de la empresa Zeuz sistemas, utilizando etiquetas RFID (Radio Frecuencia e Identificación) para la identificación de ítems y su geolocalización.

b. Objetivos Específicos

- Analizar el proceso de información que interviene en el manejo de inventarios y control de stock dentro de la empresa.
- Utilizar una metodología de desarrollo de software que permita el desarrollo modular, personalizado y con la participación activa del cliente.
- Aplicar el protocolo RFID de comunicación para controlar mediante un ordenador el dispositivo de lectura y escritura de las tarjetas de radiofrecuencia.
- Implementar un mecanismo de geolocalización de los productos utilizando protocolos de ubicación geográfica a través del internet.
- Aplicar protocolos y estándares que permitan intercambiar datos entre diferentes aplicaciones y plataformas.

1.5 ALCANCE

El sistema garantizará la calidad de los procesos de control de stock y la gestión del inventario de los productos dentro de la empresa ZEUZ SISTEMAS, utilizando etiquetas de radiofrecuencia, teniendo en cuenta que se utilizará la intranet proporcionada por la empresa.

El sistema contará con dispositivos de lectura y escritura para controlar los tags RFID, dentro de la intranet de la empresa, para gestionar óptimamente los bienes y

realizar el proceso de inventarios utilizando tecnología RFID dentro de un rango de lectura acorde con las capacidades de los dispositivos que la empresa provea, el sistema se compondrá de distintas fases como se detalla a continuación.

- Módulos del Sistema.

El sistema permitirá la gestión del inventario, control del stock y geolocalización de los productos, los mismos que tendrán las siguientes funcionalidades.

- Gestión de Autenticación de Usuarios.
- Gestión de Roles de Usuarios
- Gestión de productos
- Gestión de bodegas
- Gestión de Proveedores
- Gestión de Familias
- Gestión de Unidades
- Gestión Control de Stock e Inventario.
- Gestión de Asignación códigos RFID y Geo localización.
- Gestión de Kardex.
- Gestión de Reportes.

La gestión comprende la ejecución de las operaciones de un CRUD (Create, Read, Delete and Update) en cada uno de los elementos previamente definidos.

- Funcionamiento y componentes.

- Los Programadores: escriben información en las etiquetas (tags o transpondedores) y estas serán asignadas a los productos pertinentes.
- Al momento de gestionar el inventario se utilizará Lectores (interrogadores) que leen la información de las etiquetas, proporcionándoles energía para que éstas transmitan la información pregrabada. Este proceso se realiza gracias a la utilización de un Middleware que permite gestionar el intercambio de datos con el Sistema de información el mismo que se procesa, para que posteriormente el administrador tome las decisiones oportunas sobre el stock e inventario de la empresa.

- Seguridad, confidencialidad y privacidad
 - Al no tener que manipular los datos que se encuentran almacenados en los microchips internos de los tags, la información se lee directamente al sistema esto proporcionará seguridad, confidencialidad y privacidad al momento de gestionar el inventario. Los tags cuentan con un código único que no permite la duplicidad de los mismos, también otra característica importante es que los tags cuentan con dos claves almacenadas en su microchip el cual permite que la información no sea modificada ni leída por otro recolector que no posea dicha clave, lo que le hace muy seguro.

- Programación del sistema
 - El sistema tendrá la capacidad de interactuar con los equipos RFID, a través de un Middleware el que nos permitirá la comunicación entre el sistema y el equipo. Este middleware será gestionado por una plataforma de programación permitiéndonos personalizar un sistema de inventarios acorde con las necesidades de la empresa.

- Implementación con la base de datos
 - Se diseñará un modelo de base de datos que permitirá tener consistencia de los datos del inventario, esta base de datos estará acorde con las necesidades de la empresa.

1.6 FACTIBILIDAD

1.6.1 Factibilidad técnica

Para el desarrollo del presente sistema se utilizará software propietario y libre, para la parte de la infraestructura física se identificará los recursos necesarios que permitirá cumplir con los objetivos.

1.6.2 Requerimientos de hardware

Los requerimientos mínimos de hardware que se aplicarán en el desarrollo del sistema son los siguientes:

- Servidor: procesador Intel Xeon CPU, Graficas GHz, 4 de RAM, Monitor de 20'' plug and play
- Computadoras personales para el desarrollo del sistema, Samsung r440 core i3
- Lectores y escritores de radio frecuencia que trabajen a 13.55 MHZ.
- Tags de radio frecuencia.

1.6.3 Requerimientos de software

Los requerimientos de software a utilizar son los siguientes:

- Sistema operativo Windows 7.
- Visual Studio 2013.
- Netbeans.
- MYSQL.
- StarUML.
- Power Designer.

1.6.4 Factibilidad Económica

El presupuesto estimado para el sistema se muestra en las siguientes tablas No 1, 2, 3,4.

Tabla 1
Descripción de Hardware

No	Material	Valor	Valor Final x Auspicio
1	Servidor	\$1.600.00	\$0.00
2	PCS	\$1.700.00	\$0.00
1	Impresora	\$200.00	\$0.00
1	Reader / Writer ACR122U	\$60.00	\$0.00
50	Tags RFID 13.56 Mhz	\$35.00	\$0.00
1	Mini Rfid 13.56 Mhz Proximidad Inteligente Usb Ic Lector	\$20.00	\$0.00
	Subtotal	\$3.615.00	\$0.00

Tabla 2
Descripción de Software

Material	Valor
Mysql	\$0.00
Netbeans	\$0.00
Power Designer (Versión estudiante)	\$0.00
StarUml	\$0.00
Primefaces	\$0.00
Visual Studio 2013	\$458.99
Subtotal	\$458.99

Tabla 3
Descripción de Recursos Humanos

Desarrollador e Arquitecto de software	Valor
Franz Gualoto Suárez	\$1000.00
Ruben Ortega Argüello	\$1000.00
Subtotal	\$2000.00

Tabla 4
Descripción de Suministros de Oficina

Material	Cantidad	Valor	Valor Auspiciante
Resma Papel Bond	2	\$6.80	\$0.00
Toners Impresora	4	\$240.00	\$0.00
Internet	6 meses	\$300.00	\$0.00
Varios		\$30.00	\$0.00
Subtotal		\$576.80	\$0.00

1.6.5 Factibilidad Operativa

La empresa Zeuz sistemas cuenta en la actualidad con bodegas de grandes capacidades para el almacenamiento físico, y personal capacitado para la toma y la digitación de información de cada uno de los artículos, pero a la hora de ingresar demasiada información colapsa y se tiende a generar datos erróneos, ya que cuenta con una base de datos digital en Excel y un sistema básico que solo puede ser operado por una persona a la vez en las diferentes áreas de trabajo.

La creación de este sistema software permitirá trabajar de manera adecuada a todo el personal y con una gran reducción de inconvenientes como la manipulación de información errónea, o duplicar artículos, así se disminuirá el tiempo en las distintas áreas de trabajo en la empresa Zeuz Sistemas.

1.6.6 Factibilidad Operacional

El sistema realizara la asignación de roles de acceso a los usuarios asegurando el correcto funcionamiento en distintas áreas de trabajo de la empresa, y cada uno de

ellos tendrán una interfaz amigable, logrando así la optimización en los procesos del sistema.

1.6.7 Factibilidad legal

En el desarrollo del sistema se utilizará herramientas Open Source durante el ciclo de vida del software, teniendo viabilidad legal positiva ya que desde el punto de vista de propiedad intelectual se podrá utilizar dichos programas sin costo alguno, se ha corroborado que todos los programas a ser utilizados no presenten ningún impedimento jurídico o alguna aporte legal dentro del país, por lo tanto no existe responsabilidad alguna para el funcionamiento de los mismos.

CAPÍTULO II

MARCO CONCEPTUAL

2.1 KARDEX

El kardex se define como los registros o documentos administrativos de control organizados de los bienes o productos de una empresa, existen varios tipos de kardex pero se enfocará en los kardex de inventarios que permite tener reportes con información concreta acerca de las transacciones del inventario de la empresa, también se puede realizar un seguimiento de los costos de los productos en los almacenes. La importancia de los kardex en los inventarios radica en que proporcionan información y ayudan en el control de los mismos, pero se debe tener bien claro el concepto de inventarios y una buena clasificación de los mismos para la aplicación en las empresas.

(Roncancio M, Cuevas , Rodríguez, Villalba , & Aguirre , 2011)

Inventarios.

El inventario es el registro documental de los bienes y objetos pertenecientes a una persona física y que se efectúa a partir de mucha precisión y prolijidad en la plasmación de los datos. También se define como inventario a la comprobación y recuento, tanto cualitativo como cuantitativo de las existencias físicas con las teóricas que fueron previamente documentadas, existen varios tipos de inventarios y se definen a continuación.

- **Inventarios de materias primas:** son bienes adquiridos para una transformación y crear otro tipo de producto terminado.
- **Inventarios de productos en proceso:** son bienes que no están listos para la venta pero ya sufrieron una transformación.
- **Inventarios de productos terminados:** son aquellos productos que fueron transformados completamente y están listos para la venta.
- **Inventarios de mercaderías no fabricadas por la empresa:** son bienes terminados pero adquiridos a otras empresas y están listas para la venta.

- **Inventarios de materiales, repuestos y accesorios:** son bienes que no son para ser vendidos ni transformados, sino que ayudan a cumplir con la razón social de la empresa.
- **Inventarios de envases y empaques:** son bienes que sirven como empaque y/o envase para los productos terminados de la empresa.
- **Inventarios en tránsito:** son productos que no se encuentran en las bodegas de la empresa pero pertenecen a la misma, estos inventarios se utilizan para la compra de productos importados, pero cuando llegan a la empresa pasan a otro tipo de inventario.

Los inventarios dependen de la empresa ya que según su razón social se pueden clasificar a sus bienes o productos de diferente manera, pues lo que para una empresa puede ser un producto terminado, para otra puede ser mercaderías no fabricadas por la empresa.

Para la gestión de los inventarios a través del kardex se requiere utilizar un método y un sistema de administración.

Sistemas de Administración de inventarios.

- Sistema periódico.
- Sistema permanente.

En los sistemas periódicos se ejecutan en lapsos de tiempos periódicos mientras que los sistemas permanentes se realizan en lapsos de tiempo muy cortos e inclusive de manera cotidiana para un mayor control.

Métodos de Administración de inventarios.

- P.E.P.S.
- U.E.P.S.
- Promedio ponderado.

Método P.E.P.S.: consiste en dar salida del inventario aquellos productos que ingresaron primero, quedando en los inventarios los productos que se compraron recientemente.

Método U.E.P.S.: al contrario del método P.E.P.S., consiste en dar salida de los inventarios a los productos que se adquirieron recientemente, quedando en los inventarios los productos antiguos, también se le conoce como últimos en entrar primeros en salir.

Método del promedio ponderado: consiste en determinar un promedio, sumando los valores existentes en el inventario con los valores de la nueva compra, para luego dividirlo entre el número de existencias en el inventario.

(Roncancio M, Cuevas , Rodríguez, Villalba , & Aguirre , 2011)

2.2 IDENTIFICACIÓN AUTOMÁTICA POR RADIO FRECUENCIA

En los últimos años se ha incrementado el uso de métodos de identificación automática en las empresas tanto nacionales como internacionales, muchos de sus procesos de identificación proporcionan información automática sobre gente, bienes, mercancías y productos, a través de la asignación de códigos únicos o el almacenamiento de datos dentro de los microchips que tienen los tags Rfids.

Las etiquetas de RFID (Radio Frecuencia) llamadas tags, provocaron una revolución en sistemas de identificación desde hace un buen tiempo, por su capacidad de almacenar información, transferencia de datos sin contacto físico entre dispositivos, es mucho más flexible y proporciona un ahorro significativo de tiempo.

2.2.1 Aplicaciones Radio Frecuencia

Este tipo de tecnología se implementa en aplicaciones como en tarjetas identificadoras sin contacto como cartas, paquetes de correos o agencias de transporte se tiene comprobado que lectura por radio frecuencia ha mitigado problemas futuros dentro de sus puestos empleados, como por ejemplo marcar

productos de un supermercado y al momento de su compra identificarse de manera sencilla a la compra.

2.2.2 Ventajas de Radio Frecuencia

El sistema más utilizado hasta ahora es el código de barras aunque tengan muchas limitaciones como integridad física, adulteración de códigos y sin capacidad de almacenamiento de datos, por tal razón se presenta a continuación las ventajas de las etiquetas de radio frecuencia que son las siguientes: (Acevedo, 2004)

- A diferencia del código de barras, las etiquetas electrónicas no necesitan contacto visual con el módulo lector para que éste pueda leerlas. La lectura se puede hacer a una distancia de hasta 10 metros. (Acevedo, 2004)
- La tecnología RFID tiene la capacidad de leer múltiples etiquetas electrónicas simultáneamente, mientras que el barcode, tienen que ser leídos secuencialmente. Esta característica de auto identificación por radiofrecuencia ofrece múltiples ventajas como, por ejemplo, la reducción del tiempo de espera de los usuarios. (Acevedo, 2004)
- Las etiquetas electrónicas poseen la capacidad de almacenar información en su microchip interno mientras que el código de barras, solo puede contener un código y, en el mejor de los casos, un precio o cantidad. (Acevedo, 2004)
- Mientras que sobre el código de barras se puede escribir solo una vez, sobre las etiquetas electrónicas se puede escribir todas las veces que haga falta. (Acevedo, 2004)
- La tecnología RFID evita la adulteración de datos. Es así que en el código de barras se puede falsificar con una simple fotocopia, mientras que en los tags Rfid, no se pueden fotocopiar y en caso de los transponder de solo lectura no tienen la opción de modificar. Un tag asignado a un artículo garantiza en su totalidad su autenticidad. (Acevedo, 2004)
- Una falencia del código de barras es que se estropea o se rompe fácilmente, mientras que, una etiqueta tag es más resistente porque, puede formar parte del producto o se recubre con una superficie protectora sin

afectar a la etiqueta y soporta mejor la humedad y la temperatura. (Acevedo, 2004)

2.2.3 Características

La tecnología RFID (Radio Frecuencia), tienen más prestaciones sobre el código de barras, ya que cuenta con mayores características como seguridad, confiabilidad y no requiere de línea de vista para su identificación, como se puede observar en la Figura 1.

Código de Barras	Radiofrecuencia
Definición	
- Es una herramienta que permite capturar datos e identificar productos en la red de valor (GS1 Colombia, 2008a).	- Es una tecnología que usa ondas de radio para identificar productos de forma automática por medio de Tags (GS1 Colombia, 2008a).
Aplicaciones e impactos en la cadena de suministro	
- Administración de inventarios.	- Trazabilidad y visibilidad de productos.
- Identificación de productos en procesos logísticos.	- Actualización de inventarios e ubicaciones en tiempo real (Branch, 2008).
- Gestión ubicaciones en centros de almacenamiento.	- Administración y control de actividades de transporte.
- En la gestión del transporte para identificar y registrar las cargas que esta moviliza.	- Mejora los flujos de información en la cadena de suministro.
Ventajas y desventajas	
- Invariabilidad de la información contenida en la etiqueta de código de barras.	- Mayor capacidad de almacenamiento de datos respecto al código de barras.
- Rango de lectura limitado.	- La información de las etiquetas puede ser variable y reutilizable.
- Necesidad de un operador para la lectura de los códigos (Jung <i>et al.</i> , 2007).	- Identificación simultánea de productos.
- Costos de implementación más bajos respecto al RFID.	- No requiere operario para lectura (GS1,2008a).
- Actualmente posee mayor confiabilidad en las lecturas de identificación de productos que el RFID.	- Las actualizaciones de inventario y ubicaciones se realizan en tiempo real.
	- Mayor capacidad de trazabilidad (Hines, 2004).
	- Reducción de costos e incremento en la exactitud y eficiencia en las operaciones de manipulación e identificación de productos (Singer, 2006).
	- Problemas de confiabilidad de lectura por lo nuevo de la tecnología (Glover y Bhatt, 2006).
Hardware**	
Etiqueta, lector o escáner, impresora.	Etiquetas y/o tags, lector o reader, antenas, impresora.
Software**	
Sistemas de información como WMS, ERP, TMS, entre otros.	- Sistemas de información como WMS, ERP, TMS, etc.
	- Middleware o decodificador.
Implementación	
- Comienza con la obtención del prefijo de la compañía en GS1 (GS1, 2008a).	- Identificación de los niveles de eficiencia de los procesos.
- Se determina la simbología de los códigos a emplear según la aplicación y tipo de producto.	- Determinación de tipo de infraestructura y recursos para el desarrollo de sistema de RFID (Schuster, Allen y Brook, 2006).
- La empresa asigna los códigos a los productos.	- Selección del tipo de tags, antenas, lectores y sistemas de información.
- Se selecciona el sistema de impresión de etiquetas y ambiente de escaneo.	- Pruebas pilotos e implementación definitiva del sistema.
- Se verifica calidad y ubicación del código en el producto (Palmer, 1995).	
Evaluación de la inversión	
- Se emplean técnicas como el ROI (<i>Return Over Investment</i>) o retorno de la inversión apoyado de la evaluación del valor presente neto.	
- Las medidas permiten evaluar el impacto del sistema en el desarrollo de las actividades logísticas de la empresa y el grado de recuperación de la inversión (Jung <i>et al.</i> , 2007).	

Figura 1. Análisis comparativo entre código de barras y Rfid

Fuente: (CORREA ESPINAL, ÁLVAREZ LÓPEZ, & GÓMEZ MONTOYA, 2010)

MARCO TEÓRICO

2.3 RFID (RADIO FRECUENCIA E IDENTIFICACIÓN)

La tecnología RFID (Radio Frecuencia e Identificación) consiste en aplicar la radio frecuencia para la identificación de objetos por medio de ondas a distancia. Un sistema de identificación por radio frecuencia en la actualidad está a la vanguardia de la tecnología para la identificación de objetos a distancia sin necesidad de contacto y visualización directa con el objeto, este sistema está conformado por etiquetas o tags y el lector. Las etiquetas o tag RFID (Radio Frecuencia e Identificación) tienen un microchip y una antena que a través de una antena de radio frecuencia se podrá identificar cada elemento que porte la etiqueta, la misma que tiene la capacidad de 2kbytes de almacenamiento de datos. El lector es capaz de leer datos almacenados en cada etiqueta, y por lo general se tiene un dispositivo que tenga una o más antenas que emitan ondas de radio y que reciban las señales de cada uno de los tags. A continuación se representa el esquema del funcionamiento básico RFID, tal como se muestra en la Figura 2.



Figura 2. Funcionamiento Sistema RFID

Fuente: (Seguros, 2010)

2.3.1 COMPONENTES.

La tecnología RFID (Radio Frecuencia e Identificación) tiene tres componentes los cuales se describen a continuación:

- **Etiqueta electrónica:** Este componente tiene la capacidad de almacenamiento de datos y existe varios tipos de memoria como son de lectura y escritura / lectura, tal como se muestra en la figura 3.

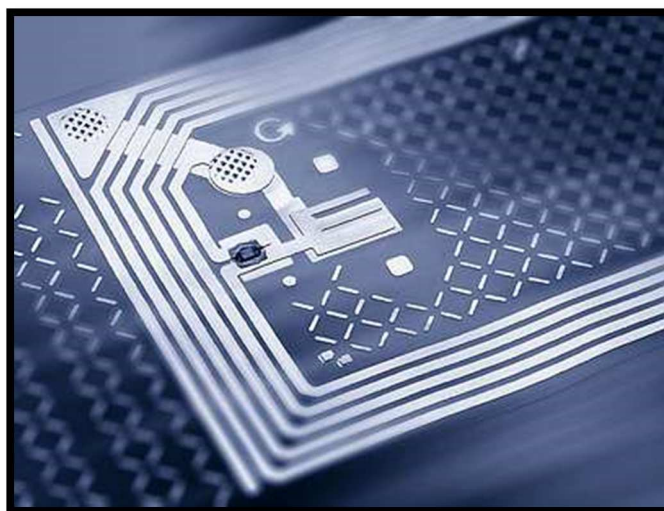


Figura 3. Diseño de dos tipos de etiquetas
Fuente: (S.A., 2015)

- **La antena:** que incorpora en cada una de los tipos de etiquetas suele ser capaz de transmitir los datos almacenados en el microchip y pueden ser de dos tipos:
 - Bobina
 - Dipolo
- **Lector:** Este componente está compuesto por una antena, un transceptor y un decodificador, que tiene la capacidad de enviar señales continuamente para revisar si existe alguna etiqueta en el lugar que se encuentre, y cuando se haya captado la señal de cada etiqueta se extrae los datos almacenados y se pasa a un sistema de procesamiento de datos.
- **Base de datos:** permite almacenar, procesar y la correlación de la información proporcionada por la etiqueta.

2.3.2 TIPOS DE ETIQUETAS

Una etiqueta o un tag está compuesto de un inductor o antena y de un circuito integrado de diferentes características, como se muestra en la figura 4.

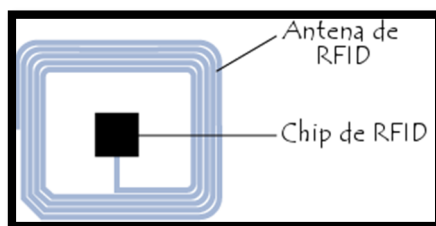


Figura 4. Composición de una tarjeta
Fuente: (Benchmark, 2015)

2.3.2.1 ETIQUETAS POR ACCESO

Este tipo de etiquetas pueden realizar las siguientes funciones para cada sistema como son:

- Solo lectura de etiquetas
- Escritura una sola vez y múltiples lecturas en las etiquetas
- Escritura y lectura múltiple.

2.3.2.2 ETIQUETAS POR MEMORIA

Este tipo de etiquetas se puede clasificar por el rango de memoria que puede procesar y ser utilizada en bits.

- Solo tag ID (64 / 96 bits)
- Tag ID (64 / 96 bits) + memoria usada en ejecución
- 0-8kbits según el fabricante

2.3.2.3 ETIQUETAS ALIMENTACIÓN

Las etiquetas de alimentación constan de tres tipos los cuales permiten transferir desde el lector a la antena de la etiqueta según la potencia que se haya implementado según la propagación que se haya utilizado en el sistema.

2.3.2.4 ETIQUETAS ALIMENTACIÓN PASIVAS

Este tipo de etiquetas RFID (Radio Frecuencia e Identificación) no tiene alimentación propia, y funciona sin una fuente o batería interna por lo que la energía requerida la obtiene a través de un campo eléctrico generado por el interrogador, y tienen una distancia de transmisión de seis metros, con una dimensión de 0.4 x 0.4 milímetros.

2.3.2.5 ETIQUETAS ALIMENTACIÓN SEMIPASIVAS O SEMIACTIVAS

Este tipo de etiquetas RFID (Radio Frecuencia e Identificación) son muy parecidas a las etiquetas pasivas, con la diferencia de que poseen una pequeña batería que le permite al circuito integrado estar constantemente activo, este tipo de alimentación no requiere capturar la potencia que emite la señal entrante para devolver los datos con una señal saliente, en este caso la tecnología tiene mejoras como la emisión de respuestas y alcanza una transmisión mayor alrededor de 30 metros.

2.3.2.6 ETIQUETAS ALIMENTACIÓN ACTIVAS

Este tipo de etiquetas RFID (Radio Frecuencia e Identificación) tiene alimentación propia la cual consta de una batería que provee energía internamente y refuerza a la señal que emite el lector, este tipo de tecnología alcanza mayores distancias y con una fiabilidad mayor de señales débiles y generar respuestas claras.

2.3.3 FUNCIONAMIENTO

La tecnología RFID (Radio Frecuencia e Identificación) funciona de la siguiente forma:

El lector envía una serie de ondas de radiofrecuencia conocidas como señal de interrogación. Si una etiqueta se encuentra al alcance de estas ondas de radio, es captadas por su antena y al mismo tiempo energizan el chip de la misma. La etiqueta retransmite al lector mediante ondas de radio el número de identificación único de la tarjeta y en el caso de que se haya solicitado más información que estuviese almacenado en la etiqueta, se transmite de la misma forma. Esta información es almacenada en memoria y procesada de acuerdo a los requerimientos del sistema y según se hubiese configurado, tal como se muestra en la figura 5. (SA, 2011)

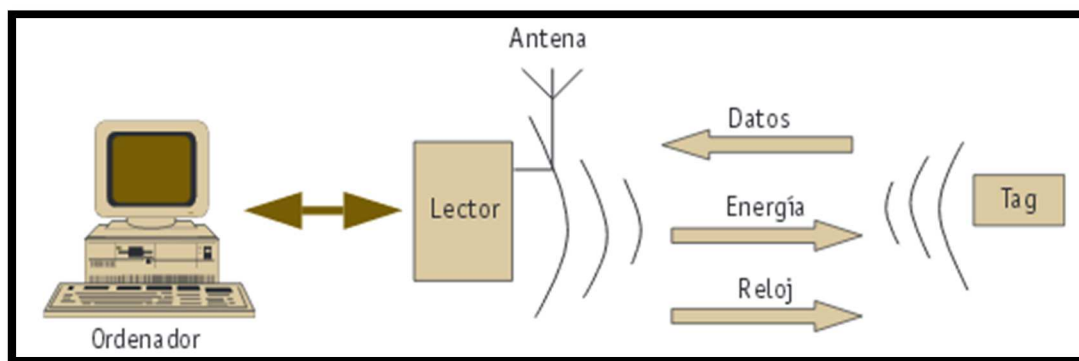


Figura 5. Funcionamiento de un sistema RFID.

Fuente: (Tecnología, 2012)

2.3.4 CONECTIVIDAD

Para el desarrollo de un Sistema de Autenticación con RFID, es necesario considerar la conectividad de los lectores, con los sistemas y tener en cuenta su capacidad de integrarse con otras tecnologías existentes. Desde el inicio de esta tecnología se han utilizado comunicaciones seriales, teniendo como opciones de conectividad las siguientes: (Pérez, 2009)

- a) **Protocolo RS-232:** comunicación de corto alcance; sus limitantes son la baja velocidad de comunicación (de 9600 bps a 115.2 kbps), y el alcance

del cable está limitado a 30 metros, falta de control de errores y que la comunicación es punto a punto (Pérez, 2009).

- b) **Protocolo RS-485:** el protocolo tipo bus permite a múltiples dispositivos estar conectados al mismo cable; maneja longitudes de cable de hasta 1200 metros y velocidades de hasta 2.5 Mbps (Pérez, 2009).
- c) **Ethernet:** se considera la mejor opción. ya que su velocidad es más que suficiente para los lectores, además de la confiabilidad de los protocolos TCP/IP sobre Ethemet. Asegurando la integridad de los datos enviados sobre una infraestructura común. (Pérez, 2009)
- d) **USB:** considerando la casi desaparición del puerto serial en las computadoras. Algunos proveedores de lectores RFID han habilitado sus equipos para comunicarse mediante el puerto USB. (Pérez, 2009)
- e) **Wireless 802.11:** esta tecnología tiene gran aceptación últimamente, ya que no requiere de cables para la comunicación (Pérez, 2009)

2.3.5 ESTÁNDARES

En la actualidad existen diferentes organismos que regulan y normalizan la tecnología RFID (Radio Frecuencia e Identificación), para que sean accesibles a través del uso de protocolos, como son la norma ISO (Organización Internacional de Estandarización), la cual define estándares comerciales, la IEC (Comisión Electrotécnica Internacional), ayuda a consolidar la cooperación internacional de las tecnologías y la electrónica.

2.3.5.1 ESTÁNDARES ISO PARA RFID (RADIO FRECUENCIA E IDENTIFICACIÓN)

En la tecnología RFID (Radio Frecuencia e Identificación) existen diversas necesidades al momento de desarrollar una aplicación, por tal motivo se requiere de normas y estándares que ayuden en el desarrollo de los sistemas que requieran de la tecnología RFID, y de esta manera tener sistemas homogéneos, a continuación se detalla las principales características de regulación que son:

- La frecuencia autorizada para el manejo de la tecnología RFID.
- La potencia que sea permitida para la aplicación generada.
- Los parámetros y métodos para la comunicación entre el lector y las etiquetas RFID.

En la actualidad la norma ISO tiene 3 estándares fundamentales para RFID (Radio Frecuencia e Identificación) que son:

- a) **ISO 14443:** esta norma se la utiliza en sistemas sin contacto, define las características hardware y la transmisión de la información, como pueden ser control de acceso, autenticación de usuarios o control de bienes.
- b) **ISO 15693:** esta norma se la utiliza para sistemas de proximidad
- c) **ISO 1800-6:** esta norma se la utiliza para especificar la interfaz aérea y las frecuencias aceptadas de forma adecuada y segura de los dispositivos.

2.3.5.2 EPCGlobal (ORGANISMO DESARROLLADOR DE ESTÁNDARES RFID)

EPC global es una organización sin fines de lucro que ha desarrollado una amplia gama de estándares para la identificación de productos. Los estándares EPC están enfocados a la cadena de suministro y particularmente definen la metodología para la interfaz aérea; el formato de los datos almacenados en una etiqueta RFID, para la identificación de un producto, captura, transferencia, almacenamiento y acceso de estos datos; así como el middleware y la base de datos que almacena esta información. (Rodas, 2009)

2.3.6 MIDDLEWARE

El sistema RFID obtiene la información de las lecturas de los Tags RFID y el middleware RFID extrae esos datos y los procesa para convertirlos en una estructura que se pueda parametrizar para posteriormente generar información que luego el Sistema de Gestión interpreta. (S.L, 2011)

2.3.6.1 FUNCIONAMIENTO DEL MIDDLEWARE

El middleware gestiona las lecturas de las antenas y esta información es procesada a través del software de gestión de la producción y del almacén. El traspaso de información se realiza directamente sobre la base de datos del sistema, resultando un procesamiento rápido y efectivo con reducción de errores. Las transacciones se realizan en tiempo real, quedando reflejadas en el sistema en el mismo instante en el que se realizan las acciones. Por ejemplo, una expedición de mercancía queda reflejada en el sistema en el mismo momento en el que los productos atraviesan el arco móvil RFID situado en la entrada del camión de transporte. (S.L, 2011)

2.3.6.2 INTEGRACIÓN DE SISTEMAS DE INFORMACIÓN CON EL MIDDLEWARE RFID.

Los sistemas RFID-EPC interactúan con dispositivos móviles, lectores, grabadores y procesadores de información, estos forman un entorno distribuido y para ser integrados en aplicaciones software se requiere de una plataforma “Middleware” el mismo que permite manejar niveles de aplicación con niveles de servicio y transporte de datos. Es decir permiten crear aplicaciones que procesan los datos independientemente del origen del mismo, en la figura 5 se representa la integración de un sistema middleware.

Es decir la aplicación gestiona la logística, inventario, stock, etc, a través del EPC como si lo hiciera con cualquier otro tipo de datos, mientras que las plataformas (Middleware) proporcionan estos EPC mediante servicios de transporte de red. Dentro de las plataformas middleware más conocidas están J2EE y .NET, tal como se muestra en la figura 6.

(M^a Victoria Bueno Delgado, Alejandro S. Martínez Sala, Esteban Egea López, Javier Vales Alonso, Joan García Haro)

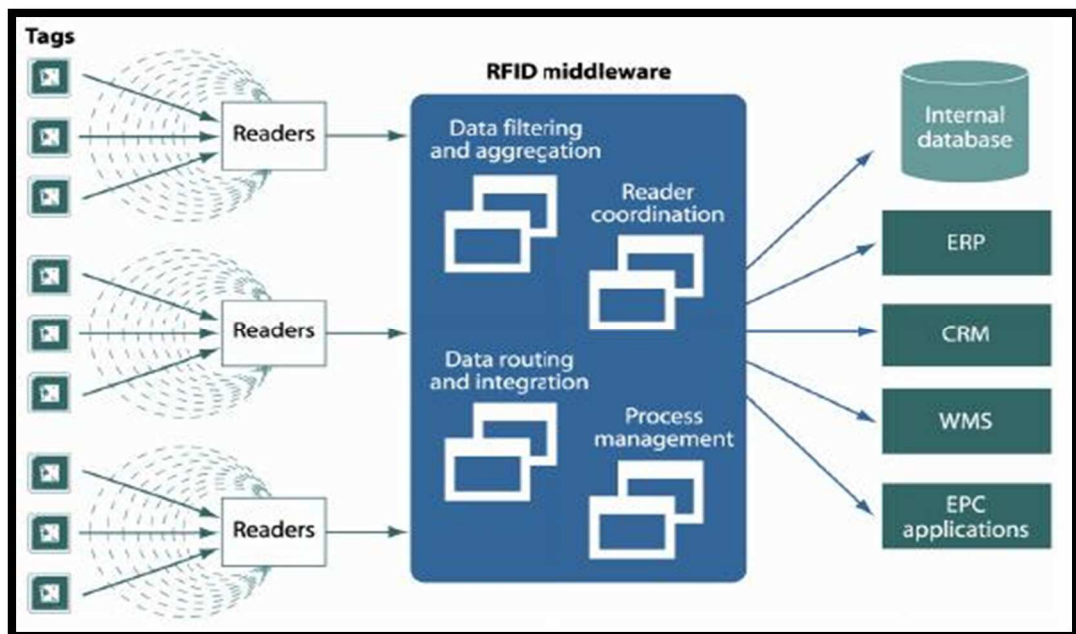


Figura 6. RFID middleware

Fuente: (© 2014 Anirban Adak. All Rights Reserved, 2014)

2.3.6.3 PLATAFORMAS MIDDLEWARE.

Muchas empresas ya se encuentran desarrollando aplicaciones RFID, y estas manejan el código EPC, a continuación se presentan algunas de ellas:

- **Sun Java System RFID**, este middleware basado en SAVANT es uno de los que más aportes ha hecho con relación a la tecnología RFID, pero tiene algunas limitaciones como por ejemplo soportan un tipo limitado de readers compatibles y manejan solo tags pasivos.
- **WinRFID**, middleware desarrollado por .NET, se define por una arquitectura multicapa que trabaja de forma independiente del hardware, soporta tags pasivos, activos, y también con readers fijos y móviles.
- **TAVIS™**, es un middleware maneja tags pasivos, activos y además maneja GPS, sistemas de localización de tiempo real y redes mesh. Una de sus ventajas es que permite la trazabilidad distribuida o centralizada.

(M^a Victoria Bueno Delgado, Alejandro S. Martínez Sala, Esteban Egea López, Javier Vales Alonso, Joan García Haro)

2.3.7 FRECUENCIAS

RFID se clasifican en frecuencias que pueden variar según el sistema que se vaya a utilizar y cada una tiene su sector de aplicación.

2.3.7.1 FRECUENCIA BAJA - LOW FREQUENCY (LF) (9 - 125 KHZ)

Su principal ventaja es su aceptación en todo el mundo, funciona cerca de los metales y está ampliamente difundida. La distancia de lectura es inferior a 1,5 metros, por lo que las aplicaciones más habituales son la identificación de animales, barriles de cerveza, auto key and lock o bibliotecas. (Electronics, 2010)

2.3.7.2 BAJAS FRECUENCIA ALTA - HIGH FREQUENCY (HF) (13,56 MHZ)

Esta frecuencia también está muy difundida, pero a diferencia de la frecuencia baja, el alta no funciona cerca de los metales. Normalmente se utiliza en aplicaciones tales como la trazabilidad de los productos, movimientos de equipajes de avión o acceso a edificios. (Electronics, 2010)

2.3.7.3 FRECUENCIA ULTRA-ALTA - UHF (433 MHZ Y 860-960 MHZ)

Los equipos que operan a estas frecuencias UHF (Ultra High Frequency) no pueden ser utilizados de forma global porque no existen regulaciones globales para su uso y su aplicación depende de la legalidad del país. Este tipo de frecuencia se usa para aplicaciones de trazabilidad con tags activos (Electronics, 2010)

2.3.7.4 FRECUENCIA DE MICROONDAS - ACTIVE FREQUENCY (2,45 GHZ Y 5,8 GHZ)

Estas frecuencias son las más habituales para los tags activos, pero no tienen el problema de la falta de regulaciones globales, además ofrecen largas distancias de lectura y altas velocidades de transmisión. Los tags activos que operan en el rango de

las microondas son muy usados para seguimiento y trazabilidad de personas u objetos (Electronics, 2010)

2.3.8 VENTAJAS

Seguridad: Es una tarjeta que por su diseño tecnológico, no puede duplicarse fácilmente. Cada una posee un código distinto y no permite que existan tarjetas duplicadas. Sin necesidad de alineación o línea vista: De todos, es el sistema más ágil y práctico, debido a que no necesita que la tarjeta sea pasada por una ranura o en el sentido correcto, lo que le da una mayor agilidad y practicidad de uso. Inventarios de alta velocidad: Múltiples dispositivos pueden ser leídos simultáneamente, esto puede ahorrar tiempo si se compara con otras tecnologías, en las que es necesario alinear los dispositivos para leerlos uno por uno. (Rodas, 2009)

2.4 GEO LOCALIZACIÓN Y RFID

La geo localización se da gracias a la integración de RFID (Radio Frecuencia e identificación), EPC (Electronic Product Code) e Internet, siendo su objetivo principal trazar y localizar geográficamente los ítems en cualquier parte del mundo o dentro de una infraestructura física.

El procedimiento consiste en adquirir la información del ítem mediante el RFID, este utiliza un formato en común definido por la EPC(Electronic Product Code), el cual estructura un código único para cada ítem basado en la información que guarda como por ejemplo, nombre del ítem, precio, caducidad, tamaño, peso, geolocalización, etc. Este formato debe ser común para todos los sistemas que requieran dicha información y a través del internet se transporta esta información a cualquier parte del mundo. En la actualidad se está remplazando los códigos de barras por RFID-EPC, ya que permiten la captura de códigos simultáneamente y traspasa materiales como madera, cartón, etc.

Dentro de las ventajas más sobresalientes se presenta:

- Automatización de todos los procesos para su almacenamiento, logística y trazabilidad.
- Los datos son detectados en tiempo real.
- Con esta tecnología quedo atrás la lectura manual (código de barras). Lo que permite reducción de tiempos y recursos.

(M^a Victoria Bueno Delgado, Alejandro S. Martínez Sala, Esteban Egea López, Javier Vales Alonso, Joan García Haro).

2.5 CARACTERÍSTICAS DEL RFID VS CÓDIGO DE BARRAS.

En la actualidad el código de barras es el método más utilizado para la identificación y ha probado ser efectivo, no obstante tiene muchas limitaciones, por lo tanto se presenta las características de Rfid vs Código de barras que se visualizarán en la siguiente Tabla 5.

Tabla 5
Tecnología RFID vs Código de Barras

Características	Tecnología RFID	Códigos de Barras
Línea de vista	No es un requisito	Debe de existir línea de vista. Los objetos deben estar separados para una lectura correcta. Muy inconveniente
Almacenamiento	Capacidad de más de 1Kb	Sin opción de almacenamiento.
Falsificación	Difícil de falsificar (puede estar escondido dentro del producto/objeto)	Fácil de falsificar, siempre expuesto en la parte exterior y por lo tanto fácil de copiar
Velocidad de proceso	Proceso automático posible	El proceso tiene que ser manual, en la mayoría de los casos. A demás de ser muy lento
Multilectura	Muchos tags pueden ser leídos al mismo tiempo	Los códigos son leídos uno a la vez
Durabilidad	Alta durabilidad	Se rompen, borran o corrompen de una manera sencilla

2.6 EMPRESA ZEUZ SISTEMAS.

Zion Sistemas, es una empresa integrada totalmente, por un grupo de profesionales ecuatorianos, con varios años de experiencia en el mundo de los sistemas de información y la utilización adecuada de la tecnología informática. (SISTEMAS, 2012)

2.6.1 MISIÓN

Crear, desarrollar y proveer soluciones informáticas empresariales integradas, con la utilización de tecnología avanzada, nuestra experiencia y de la manera más rentable a los intereses de nuestros clientes. Con esto se desea ayudar al crecimiento y cumplimiento de los objetivos de su organización. (SISTEMAS, 2012)

2.6.2 VISIÓN

Ser una empresa de desarrollo número uno a nivel nacional que garantice soluciones informáticas de alto nivel a nuestros clientes. (SISTEMAS, 2012)

2.7 METODOLOGÍA DE DESARROLLO DE SISTEMAS XP (EXTREME PROGRAMMING).

La metodología extreme programming fue desarrollada por Kent Beck en el año de 1999, su gran capacidad de adaptación a los cambios permite que sea una metodología muy flexible. Todo proyecto de desarrollo de software sufre cambios sobre los requerimientos en cualquier etapa del desarrollo de software y estos cambios ocurren con mayor frecuencia en la toma de requerimientos, especialmente al querer definirlos todos al principio y tratar de controlarlos es una tarea casi imposible. Por tal motivo la metodología XP propone ser muy ágil poniendo como punto principal la interacción entre los involucrados, todo el desarrollo del proyecto está guiado en grupos de trabajo y desarrollo, la comunicación es fluida y constante,

y las soluciones implementadas no contienen un grado alto de complejidad, tratan de ser lo más simple posible inclusive el código de programación. Extreme programming está siempre preparada para los cambios que se puedan dar en cualquier fase del desarrollo de software ya que está enfocada a proyectos con requisitos muy variantes e imprecisos. (Bolivariana, 2015)

2.7.1 PRÁCTICAS BÁSICAS DE EXTREME PROGRAMMING.

XP define 12 prácticas básicas para su buen desempeño, y hace gran énfasis en que se debe seguir a cabalidad las mismas:

- a) **Equipo Completo:** Se componen por todas las personas involucradas en el proyecto, incluido el cliente.
- b) **Planificación:** Se debe planificar en secuencia todas las actividades definidas por las historias de usuario y sus respectivos controles.
- c) **Test del cliente:** Se crean pruebas que validan las mini-versiones definidas entre el cliente y los desarrolladores.
- d) **Versiones pequeñas:** También llamadas mini-versiones, son pequeñas soluciones funcionales y útiles para los clientes ya que deben ser probados, éstas deben ser desarrolladas en cortos lapsos de tiempo (pocas semanas).
- e) **Diseño simple:** Se crea soluciones y códigos muy simples pero entendibles.
- f) **Pareja de Programadores:** Es una buena práctica trabajar en parejas las mismas que se combinan a diario.
- g) **Pruebas automáticas:** Se crean programas automáticos de prueba las mismas que deben ejecutarse con bastante frecuencia.
- h) **Interacción continua:** Es importante tener un ejecutable funcional del proyecto, y que se recompila cuando se crea una nueva funcionalidad, posteriormente hacer pruebas.
- i) **Código público:** Todos los desarrolladores deben tener acceso al código y sobre todo conocer el mismo.
- j) **Normas de codificación:** Se debe definir un formato de programación para que sea uniforme y entendible por todos.

- k) **Metáforas:** Los nombres de las partes del programa deben ser caros y que definan la funcionalidad del mismo, para que se den una idea rápida de que se trata y sobre todo que no existan confusiones.
- l) **Ritmo Sostenible:** Definir un ritmo persistente de trabajo es decir que no existan días muertos y otros días con muchas tareas, planear efectivamente las horas de trabajo. (Bolivariana, 2015)

2.7.2 FASES

XP (Extreme Programming) utiliza fases para el ciclo de vida del sistema, como se muestra en la figura 7.



Figura 7. Fases de la metodología XP.

2.7.2.1 Fase 1ª: Planificación del proyecto

a) Historias de Usuario.

Las historias de usuarios están definidas por el cliente, y especifican en tres o cuatro líneas los requerimientos en lenguaje no técnico y sin detalles, su utilización permite la estimación de tiempos en el desarrollo del aplicativo, por otra parte se las utiliza en la fase de pruebas para el control del cumplimiento de la aplicación. El tiempo aproximado de una historia de usuario es de una a tres semanas y se implementa en una reunión entre el cliente y el desarrollador para definir y verificar el cumplimiento de la historia.

b) Plan de Entregas

En este plan se especifica las fechas de entregas de las historias de usuarios para cada versión del programa, también se coordinan tiempos de implementación de las historias de usuario, la prioridad de implementación las mismas que son establecidas por el cliente y el desarrollador. Es importante tener en cuenta el cumplimiento de los objetivos, el número de personas involucradas en este proceso y el control de la calidad del trabajo realizado.

c) Velocidad del proyecto

Es una métrica que determina la velocidad con que se desarrolla el proyecto y se determina a través del número de historias de usuarios implementadas en una iteración. Al usar ésta métrica se podrá controlar que todas las tareas se cumplan en el tiempo predispuesto en cada iteración, es importante reevaluar cada tres o cuatro iteraciones, en el caso de no poder cumplir con los tiempos se pueden renegociar con el cliente.

d) Iteraciones

Las iteraciones definidas por XP son de tres semanas de duración, al comienzo de la iteración el cliente prioriza las historias de usuario a implementarse definidas en el plan de entregas, posteriormente las historias son divididas en tareas y asignadas a los programadores, estas tareas tendrán una duración de uno a tres días.

e) Rotaciones (Programación en parejas)

El objetivo de la programación en pareja es aumentar la calidad del desarrollo y la productividad, el mismo que consiste en reunir dos programadores y el uno analiza los métodos y funciones mientras que el otro codifica. También fomenta el trabajo en equipo, el crecimiento del conocimiento de los programadores y la optimización del código.

f) Reuniones

XP recomienda que se realice reuniones diarias donde se expongan todas las inquietudes, problemas o dudas que se tengan sobre el proyecto y de manera conjunta aportar con soluciones a los mismos, es importante que todos los involucrados aporten con ideas y que sean tomadas en cuenta.

2.7.2.2 Fase 2ª: Diseño

a) Metáfora del sistema (Glosario)

Es importante definir un glosario de términos y la utilización de nombres coherentes que permitan identificar con una palabra lo que hace una función, método o clase, de ésta manera será más sencillo de entender y reutilizar el código.

Tarjetas CRC (Clase – Responsabilidad - Colaborador)

Estas tarjetas definen una programación orientada a objetos, en la parte superior se escribe la clase a la que pertenece el objeto, a la izquierda en una columna se escriben las responsabilidades que debe cumplir los objetos y a la derecha todas las clases requeridas por cada responsabilidad.

b) Riesgos

Una pareja de desarrolladores investigan y reducen los riesgos que supone un problema potencial que surge durante el diseño.

c) Funcionalidad mínima

Requiere de soluciones simples y sencillas para poder entenderlos sin mayor esfuerzo, desarrollarlo en el menor tiempo posible y con un bajo esfuerzo y también poder reutilizarlo.

d) Re factorizar

Permite optimizar el código, sin cambiar el funcionamiento, este proceso modifica la estructura y el código.

2.7.2.3 Fase 3ª: Codificación.**a) Disponibilidad del cliente**

El cliente es fundamental en la fase de codificación porque él crea las historias de usuario y determina el tiempo de su implementación. También estará presente en las especificaciones detalladas de las historias de usuario, de su funcionamiento, y de las pruebas para verificar su correcta implementación.

b) Utilidad de Pruebas

Es importante crear pequeños test que permitan saber si el código implementado cumple con las historias de usuario. Esto se ejecutan por cada unidad desarrollada y de ésta manera se sigan sumando las unidades hasta que el desarrollo cumpla con todos los requerimientos.

c) Programación por parejas

XP, sugiere tener un repositorio de código fuente, donde sea guardada las implementaciones por los programadores cada determinado tiempo

d) Integración

Debe existir un desarrollo en equipo o colectivo donde todas las tareas sean conocidas por los desarrolladores, es decir todos tendrán la capacidad de entender y modificar cualquier código y subir al repositorio.

También es importante que el desarrollo sea escalable, es decir que se vaya agregando funciones al sistema previamente validados en su funcionamiento.

2.7.2.4 Fase 4ª: Pruebas

a) Implantación

Todo el desarrollo (código) se podrá implantar siempre y cuando haya pasado todos los test y su correspondiente validación de funcionamiento.

b) Pruebas de Adaptación

Para este proceso se realizan dos tipos de pruebas, el primero llamado test de funcionamiento que es para la verificación de las versiones, controla cada historial y su funcionamiento, y la segunda prueba llamada test de aceptación el cual se realiza en conjunto con el usuario o cliente para verificar su correcto funcionamiento para su posterior implementación.

A continuación se presenta un esquema de las fases de la metodología de desarrollo de software Extreme Programmig XP, tal como se muestra en la Figura 8.

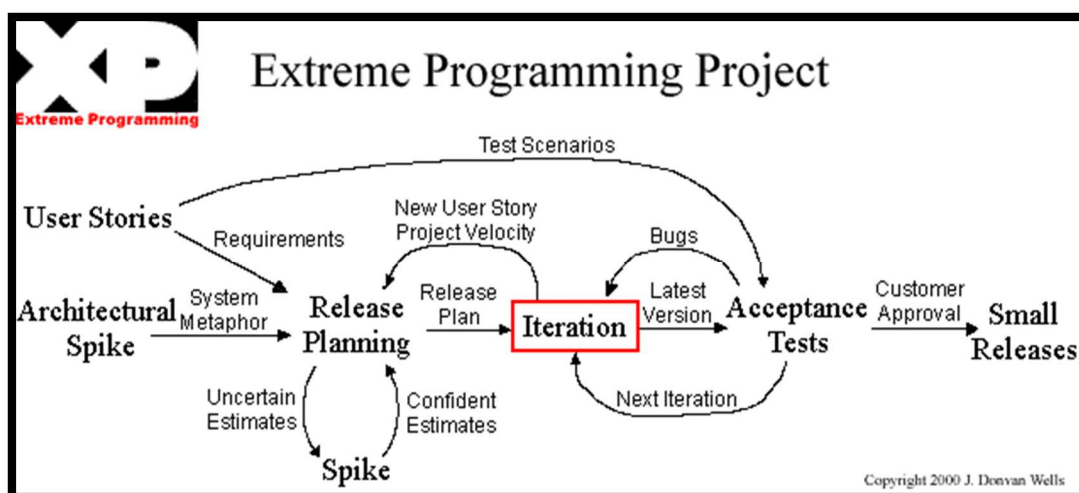


Figura 8. Esquema de las fases de XP.
Fuente: (OFBiz, 2014)

2.8 HERRAMIENTAS DE DESARROLLO.

Las herramientas de desarrollo que están acorde con las necesidades del proyecto según su capacidad de almacenamiento, procesamiento y tiempos de respuesta se describen a continuación:

Descripción de gestores de base de datos sus características, ventajas y desventajas se observan en la Figura 9.

Herramientas de desarrollo de software se describen en la Tabla 6, sobre Lenguajes de Programación cuadro comparativo, el cual permite describir los IDEs tanto propietarios como Open Source.


				
EMPRESA	Oracle Corporation	Sun Microsystem	Microsoft	PostgreSQL Global Development Group
LICENCIA	Privada.	Libre a nivel de usuario, pero para las empresas que quieran incorporarlo en productos privativos deben comprar la licencia.	Privada	Libre
VENTAJAS	<ul style="list-style-type: none"> ✓ Oracle es el motor de base de datos relacional más usado a nivel mundial. ✓ Puede ejecutarse en todas las plataformas, desde un PC hasta una supercomputadora. ✓ Oracle es la Base de datos con mas orientación hacia INTERNET ✓ Soporte de transacciones ✓ Estabilidad ✓ Soporte multiplataforma 	<ul style="list-style-type: none"> ✓ No necesita mucha memoria ram. ✓ Su bajo consumo lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema ✓ Fácil configuración e instalación. ✓ Múltiples motores de almacenamiento. ✓ Agrupación de transacciones. ✓ Replicación segura. ✓ Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está ✓ Planificación de eventos. ✓ Conectividad segura. ✓ Búsqueda e indagación de datos. ✓ Buenas utilidades de administración. ✓ Buena integración con php. 	<ul style="list-style-type: none"> ✓ Gran facilidad de configuración e instalación. ✓ Utiliza una extensión al SQL estándar, que se denomina TransactSQL ✓ Seguridad: SQL permite administrar permisos a todo. ✓ Permisos a nivel de servidor, seguridad en tablas, permitieron lectura, etc ✓ Ofrece una potente forma de unir SQL e Internet. ✓ Consultas jerárquicas con select from 	<ul style="list-style-type: none"> ✓ Sistema gratis ✓ Conexión estable. ✓ Permisos a nivel de columna. ✓ Consultas complejas ✓ Integridad transaccional Control de Concurrencia (multiversión) ✓ El código fuente está disponible para todos.
DESVENTAJAS	<ul style="list-style-type: none"> * El mayor inconveniente es el precio del producto y de su licencia. * Un Oracle mal configurado es potencialmente lento 	<ul style="list-style-type: none"> * No tiene soporte. * No permite el modo de autenticación local * No sincroniza los datos con otras bases de datos réplicas. 	<ul style="list-style-type: none"> * Requiere una enorme cantidad de memoria RAM para la instalación y utilización del software. * La relación calidad-precio esta muy debajo comparado con Oracle. 	<ul style="list-style-type: none"> * Requiere administradores capacitados. * Lento en comparación con mysql o sql server

Figura 9. Cuadro comparativo de base de datos MySQL y SqlServer.

Fuente: (Hernan, 2012)

Tabla 6
Lenguajes de Programación – cuadro comparativo

Lenguaje	¿Qué es?	Ventajas	Desventajas	Sistema Operativo
C#	Es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma net. Es un IDE propietario.	La plataforma .Net Sirve para hacer aplicaciones de escritorio, aplicaciones web y móviles.	Se tiene que conseguir una versión reciente de visual studio .net, por otra parte se tiene que tener algunos requerimientos mínimos del sistema para poder trabajar adecuadamente.	Sistema operativo Windows
JAVA	Es un lenguaje orientado a objetos, de una plataforma independiente, fue desarrollado por la compañía SUN Microsystems ahora es propietario ORACLE. Java Platform, Standard Edition o Java SE Java Platform Enterprise Edition o Java EE Java Platform Micro Edition o Java ME	Se pueden realizar distintos aplicativos, como son applets, aplicaciones de escritorio que se ejecutan en forma independiente, etc. Se puede realizar soluciones empresariales en un entorno web Soporta el desarrollo de aplicaciones móviles	Esperar la actualización siguiente para que sea más rápido.	Sirve para todos los sistemas operativos y si no es la versión adecuada para dicho sistema, la misma aplicación java se encarga de descargas o actualizar versión para un excelente desempeño en el pc. Unix, Linux, Solaris, Windows, mac.

Fuente: (Buhoos.com, 2015)

2.9 Herramienta a utilizar

En el presente proyecto de desarrollo de software es importante indicar que herramienta se utilizará en el proceso de codificación de código durante todo su ciclo y el tiempo estimado.

La herramienta a utilizar será Java por la alta referencia que se conoce y la necesidad que tiene el proyecto el cual será orientado a la web y a su vez el lenguaje de programación mantiene una robustez a cada una de sus aplicaciones como es el patrón de diseño MVC (Modelo Vista Controlador), como se muestra en la tabla N6 una comparación entre lenguajes de programación más utilizados para distintas áreas como son web e desktop.

Frontend: Se utiliza JSF, CSS3 y primefaces. Desde el Frontend, es posible hacer varias peticiones a los artículos mediante los tags.

Backend: MySQL.

Servidor de aplicaciones: GlassFish Server será usado principalmente para enviar páginas web, las aplicaciones web son diseñadas o utilizan características propias de este servidor web.

Herramienta Case: Start UML, Power Designer, serán utilizadas para el desarrollo de los diagramas ULM que sean necesarios en el ciclo de vida del software, y el modelado de la Base de Datos tanto modelo conceptual como físico.

CAPÍTULO III

ANÁLISIS Y DISEÑO DEL SISTEMA

3.1 ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

3.1.1 INTRODUCCIÓN

Este documento es una especificación de requerimientos de Software (ERS) del sistema informático basado en tecnología RFID (radio frecuencia e identificación) y geolocalización para el control del stock y gestión del inventario por lotes, para la empresa Zeuz sistemas contiene el resultado de la especificación de casos de uso del sistema el cual permite mitigar algunas falencias de codificación de la información.

3.1.2 Propósito

El presente documento tiene el propósito de definir los requerimientos funcionales y no funcionales que debe tener el sistema informático basado en tecnología RFID (radio frecuencia e identificación) y geolocalización para el control del stock y gestión del inventario por lotes, para la empresa Zeuz, buscando la realización de estas actividades.

3.1.3 Alcance

El presente proyecto está dirigido a la empresa Zeuz Sistemas, este documento presenta procesos que se toman a consideración para el desarrollo del sistema los cuales han ido generando documentación producto de la las reuniones mantenidas con los involucrados.

Específicamente lo que se espera lograr con el sistema es lo siguiente:

- Se espera ingresar todas las partes de computadoras.
- Gestionar todos los clientes de la empresa Zeuz Sistemas.

- Gestionar todos los proveedores de la empresa.
- Proporcionar roles de usuario tanto para administrador, jefe de almacén, recolector o usuario.
- Lectura y escritura de artículos por medio de Radio frecuencia.
- Generar solicitudes de nuevos productos.
- Generar solicitudes de nuevos proveedores.
- Gestionar el stock de la empresa Zeuz Sistemas.
- Gestionar solicitudes de compra.
- Gestionar Kardex de la empresa Zeuz Sistemas.
- Generar informes de productos en existencia y devaluados.
- Gestión de inventarios tanto para lectura y el informe del estado del stock máximo o mínimo.
- Organizar físicamente cada una de las partes

3.1.4 Especificación de requisitos

La especificación de requisitos se detalla como funcionalidades que va a tener el sistema, para XP (Extreme Programing) se realiza por medio de historias de usuario.

3.1.5 Definición de acrónimos

Ver glosario

3.1.6 Referencias

- XP
- Glosario
- Entrevistas con los Involucrados

3.1.7 Resumen

El XP se centrará principalmente en la Descripción General (se describen los factores generales que afectan el servicio y los requerimientos) y en los Requerimientos Específicos (consiste en la especificación de los requerimientos a un

nivel de detalle suficiente para permitir a los diseñadores y programadores desarrollar un sistema para satisfacerlos)

3.2 Historias de usuario

3.2.1 Concepto

Una manera Simple de describir una tarea concisa que aporta valor al usuario o al negocio. No se detalla más hasta el momento que la historia de usuario se vaya a desarrollar. Las historias de usuario pueden ser creadas durante las conversaciones con los usuarios interesados (stakeholders) sobre nuevas funcionalidades o mejoras del proyecto.(Mora, 2010)

3.2.2 Desarrollo

A continuación se muestra cada una de las historias de usuario y sus respectivas funcionalidades, prioridades y riesgos, el cual se ha recopilado información necesaria durante el desarrollo del sistema conforme se desarrolla el proyecto ya que cada una de sus iteraciones que se han tenido, hasta lograr el funcionamiento necesario y optimo dentro de la Empresa, donde se definen los requerimientos solicitados por el cliente, en este caso el Señor Ing. Guillermo Narváez Gerente General de Zeuz Sistemas.

3.2.3 Historias de Usuario

En las tablas N° 7,8,9,10,11,12,13,14,15,16,17 se pueden visualizar cada una de las historias de usuario recopiladas durante el ciclo de recopilación de información, para la implementación de un sistema informático basado en tecnología RFID (radio frecuencia e identificación) y geolocalización para el control del stock y gestión del inventario por lotes, para la empresa Zeuz sistemas

Tabla 7
Historia de Usuario 1

Historia de Usuario	
Número: 1	Usuario: Administrador, Jefe Bodega, Recolector
Nombre historia: Autenticar Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega.	
<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar usuarios de forma correcta mediante un botón, al ingresar a la pantalla principal se desplegará un formulario donde se registra el Nick y la Contraseña de usuario, mediante un click se accede al sistema según su rol.</p>	
<p>Observaciones:</p> <p>Se permite acceder al sistema una vez que se encuentre registrado correctamente, y redirigirse a la pantalla que sea asignado tanto como administrador, jefe de bodega o recolector.</p>	

Tabla 8
Historia de Usuario 2

Historia de Usuario	
Número: 2	Usuario: Administrador
Nombre historia: Gestión de datos de Usuarios.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<div style="border: 1px solid black; padding: 2px; display: inline-block;"> Continua </div>	

<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar usuarios, al ingresar a la pantalla de mantenimientos se desplegará un listado de usuarios, mediante un clic en la lista, el sistema desplegará una pantalla con su respectiva nómina de roles disponibles para la asignación de manera adecuada, una vez lleno todos los campos necesarios el administrador podrá guardar la información.</p>
<p>Observaciones:</p> <p>El administrador podrá gestionar usuarios siguiendo la plantilla que se muestra en pantalla, se registrará nombre, apellido, dirección, email, teléfono y rol el cual se debe ingresar como campos obligatorios, los mismos que están marcados con asteriscos (*), caso contrario se muestra como mensajes de alerta.</p>

Tabla 9
Historia de Usuario 3

Historia de Usuario	
Número: 3	Usuario: Administrador
Nombre historia: Gestión de roles de Usuario	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 3.5	Iteración asignada: 1
Programador responsable: Franz Gualoto , Rubén Ortega	
<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar roles de usuarios, al ingresar a la pantalla de mantenimientos de roles se desplegará un listado de roles, mediante un click en la lista, el sistema desplegará su información, una vez lleno todos los campos necesarios el administrador podrá guardar la información.</p>	
<p>Observaciones:</p> <p>El administrador podrá gestionar roles de usuarios siguiendo en formato de plantilla que se muestra en pantalla, en el cual se registrará nombre, descripción, e iniciales, el cual se debe ingresar campos obligatorios, los mismos se encuentran marcados con asteriscos (*), caso contrario se muestra como mensajes de Alerta.</p>	

Tabla 10
Historia de Usuario 4

Historia de Usuario	
Número: 4	Usuario: Administrador, Jefe de Bodega
Nombre historia: Gestión de datos de Proveedor.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar Proveedores, al ingresar a la pantalla de mantenimientos donde se desplegará un listado de Proveedores, mediante un clic en la lista el sistema desplegará una pantalla con su respectiva nómina de Proveedores disponibles para la asignación de manera adecuada, una vez lleno todos los campos necesarios el administrador podrá guardar los cambios.</p>	
<p>Observaciones:</p> <p>El administrador podrá gestionar usuarios siguiendo la plantilla que se muestra en pantalla, y se registrará nombre, apellido, dirección, cédula, email, los cuales son campos obligatorios, y estarán marcados con asteriscos (*), caso contrario se muestra un mensaje de alerta.</p>	

Tabla 11
Historia de Usuario 5

Historia de Usuario	
Número: 5	Usuario: Administrador
Nombre historia: Gestión de datos de Bodega.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 4	Iteración asignada: 1

Programador responsable: Franz Gualoto, Rubén Ortega
<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar Bodegas, al ingresar a la pantalla de mantenimientos se desplegará un listado de Bodegas, mediante un clic en cualquier registro de la tabla dinámica el sistema desplegará una pantalla su respectiva nómina de Bodegas disponibles para la asignación de artículos de manera adecuada, una vez lleno todos los campos necesarios el administrador podrá guardar los cambios.</p>
<p>Observaciones:</p> <p>El administrador podrá gestionar Bodegas siguiendo la plantilla que se muestra en pantalla, y se registrará nombre, descripción, y estos campos deben ser obligatorios, los mismos que están marcados con asteriscos (*), caso contrario se muestra un mensaje de alerta.</p>

Tabla 12
Historia de Usuario 6

Historia de Usuario	
Número: 6	Usuario: Administrador
Nombre historia: Gestión de datos de Unidad.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar Unidades monetarias de forma adecuada, al ingresar a la pantalla de mantenimientos se desplegará registros previamente creadas, mediante un click en cada uno de los registro de la tabla dinámica el sistema desplegará una pantalla su respectiva nómina de Unidades disponibles para la asignación de artículos manera adecuada, una vez lleno todos los campos necesarios el administrador podrá guardar la información.</p>	
<p>Observaciones:</p> <p>El administrador podrá gestionar Unidades Monetarias siguiendo en formato de plantilla que se muestra en pantalla, en el cual se registrará nombre, el cual se debe ingresar campos obligatorios, los mismos están marcados con asteriscos (*), caso contrario se muestra como mensajes de alerta.</p>	

Tabla 13
Historia de Usuario 7

Historia de Usuario	
Número: 7	Usuario: Administrador
Nombre historia: Gestión de datos de Familia.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar Familia monetarias de forma adecuada, al ingresar a la pantalla de mantenimientos se desplegará un listado de Familias de artículos, mediante un click en la lista el sistema desplegará una pantalla su respectiva nómina de familias disponibles para la asignación de artículos manera adecuada, una vez lleno todos los campos necesarios el administrador podrá guardar la información.</p>	
<p>Observaciones:</p> <p>El administrador podrá gestionar Familias siguiendo en formato de plantilla que se muestra en pantalla, en el cual se registrará nombre, el cual se debe ingresar campos obligatorios, los mismos están marcados con asteriscos (*), caso contrario se muestra como mensajes de alerta.</p>	

Tabla 14
Historia de Usuario 8

Historia de Usuario	
Número: 8	Usuario: Administrador, Jefe Bodega
Nombre historia: Gestión de datos de Artículo.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<div style="border: 1px solid black; padding: 2px; display: inline-block;"> Continua </div>	

Descripción:

El Administrador del sistema tendrá la opción de gestionar Artículos de forma adecuada, al ingresar a la pantalla de mantenimientos se desplegará un listado de artículos, mediante un click en la lista el sistema desplegará una pantalla su respectiva nómina de familias y unidades monetarias disponibles para la asignación de artículos de manera adecuada, una vez lleno todos los campos necesarios el administrador podrá guardar la información.

Observaciones:

El administrador podrá gestionar Artículos siguiendo en formato de plantilla que se muestra en pantalla, en el cual se registrará nombre, descripción, familia, unidad el cual se debe ingresar campos obligatorios, los mismos están marcados con asteriscos (*), caso contrario se muestra como mensajes de alertas.

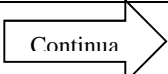
Tabla 15
Historia de Usuario 9

Historia de Usuario	
Número: 9	Usuario: Administrador
Nombre historia: Autorización de notas de ingreso.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
Descripción:	
<p>El Administrador del sistema tendrá la opción de gestionar Notas de Ingreso de forma adecuada, al ingresar a la pantalla de Verificación de notas de ingreso se necesita especificar la fecha en el cual se registró la nota de ingreso realizada por el jefe de bodega, mediante un click en buscar en el sistema desplegará una pantalla con su respectiva nómina de notas de ingreso disponibles para la aprobación adecuada de la misma, una vez aprobado, el administrador podrá guardar la información y generar reportes de las búsquedas.</p>	
Observaciones:	
<p>El administrador podrá especificar las fechas de búsqueda, siguiendo en formato de plantilla que se muestra en pantalla, donde se registrará fecha inicial e final y marcar ingreso, el cual debe ser obligatorio dentro de la búsqueda, caso contrario se muestra como mensajes de alerta.</p>	

Tabla 16
Historia de Usuario 10

Historia de Usuario	
Número: 10	Usuario: Administrador
Nombre historia: Reportes de Nota de egreso.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<p>Descripción:</p> <p>El Administrador del sistema tendrá la opción de gestionar reportes de Notas de Egreso de forma adecuada, al ingresar a la pantalla de Verificación de notas de ingreso se necesita especificar la fecha en el cual se registró la nota de egreso realizada por el jefe de bodega, mediante un click en buscar, el sistema desplegará una pantalla con su respectiva nómina de notas de egreso realizadas, el administrador podrá generar reportes de las búsquedas.</p>	
<p>Observaciones:</p> <p>El administrador podrá especificar las fechas de búsqueda, siguiendo en formato de plantilla que se muestra en pantalla, donde se registrará fecha inicial e final y marcar ingreso, el cual debe ser obligatorio dentro de la búsqueda, caso contrario se muestra como mensajes de alerta.</p>	

Tabla 17
Historia de Usuario 11

Historia de Usuario	
Número: 11	Usuario: Administrador; Jefe Bodega
Nombre historia: Control Stock Artículo	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 3.5	Iteración asignada: 1
	

Programador responsable: Franz Gualoto , Rubén Ortega
<p>Descripción:</p> <p>El usuario del sistema tiene la opción de verificar Stock de Artículos mediante la opción Stock Artículo-Bodega, se desplegará una lista de Artículos en una Tabla dinámica, el cual muestra la cantidad de stock del mismo y el usuario verificará para cada artículo que se puede satisfacer de la demanda del stock disponible en cada bodega, que será menor a 9 ítems.</p>
<p>Observaciones:</p> <p>El usuario tiene opciones búsqueda:</p> <ol style="list-style-type: none"> 1. Búsqueda por Nombre de Artículo 2. Búsqueda por nombre de familia 3. Búsqueda por su mejor criterio

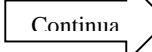
Tabla 18
Historia de Usuario 12

Historia de Usuario	
Número: 12	Usuario: Administrador; Jefe Bodega
Nombre historia: Control Stock Bodega	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 3.5	Iteración asignada: 1
Programador responsable: Franz Gualoto , Rubén Ortega	
<p>Descripción:</p> <p>El usuario del sistema tiene la opción de verificar Stock de Bodega mediante la opción Stock Artículo-Bodega, se desplegará una lista de Bodegas en un ComboBox, mediante un click se desplegará los Artículos en una Tabla dinámica, el cual muestra la cantidad de stock del mismo y el usuario verificará para cada artículo y bodega que se puede satisfacer de la demanda del stock disponible en cada bodega, que será menor a 9 ítems.</p>	
<p>Observaciones:</p> <p>El usuario tiene opciones búsqueda:</p> <ol style="list-style-type: none"> 1. Búsqueda por Nombre de Artículo 2. Búsqueda por nombre de familia 3. Búsqueda por su mejor criterio 	

Tabla 19
Historia de Usuario 13

Historia de Usuario	
Número: 13	Usuario: Administrador
Nombre historia: Control Kardex	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 3.5	Iteración asignada: 1
Programador responsable: Franz Gualoto , Rubén Ortega	
<p>Descripción:</p> <p>El usuario del sistema tiene la opción de Movimientos Kardex, se desplegará una lista de Artículos en un ComboBox, mediante un click se desplegará los Artículos en una Tabla dinámica, el cual muestra, si es ingreso o egreso, bodega del Artículo, cantidad, valor de compra, valor artículo, valor total y el stock del mismo, el Administrador verificará para cada artículo y bodega, de cada uno de los movimientos.</p>	
<p>Observaciones:</p> <p>El administrador verificará el correcto funcionamiento de cada artículo mediante un kardex.</p>	

Tabla 20
Historia de Usuario 14

Historia de Usuario	
Número: 14	Usuario: Jefe Bodega
Nombre historia: Creación notas de ingreso.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<p>Descripción:El Jefe de Bodega tendrá la opción de gestionar Notas de Ingreso de forma</p>	
	

<p>adecuada, luego se procederá a guardar los cambios mediante varias opciones:</p> <ol style="list-style-type: none"> 1. Proveedor: combobox que permite la búsqueda de proveedores 2. Bodega destino: combobox que permite la búsqueda de bodegas. 3. Artículo: combobox que permite la búsqueda de un artículo. 4. Concepto: texto que permite incluir una razón del ingreso 5. Cantidad: texto donde se ingresa la cantidad requerida. 6. Valor Artículo: texto donde se ingresa el valor de artículo unitario. 7. Agregar: Botón que permite agregar a una tabla los datos proporcionados. 8. Remover: botón que permite remover artículos de la tabla. 9. Procesar ingreso: botón que permite procesar toda la información lista del ingreso.
<p>Observaciones: El jefe de bodega podrá especificar el concepto del ingreso, el cual debe ser obligatorio dentro del proceso, caso contrario se muestra como mensajes de aletas.</p>

Tabla 21
Historia de Usuario 15

Historia de Usuario	
Número: 15	Usuario: Jefe Bodega
Nombre historia: Creación notas de egreso.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 4	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
<p>Descripción:</p> <p>El Jefe de Bodega tendrá la opción de gestionar Notas de Egreso de forma adecuada, el Jefe de Bodega podrá guardar la información, luego se procederá a guardar los cambios mediante varias opciones:</p> <ol style="list-style-type: none"> 1. Proveedor: combobox que permite la búsqueda de proveedores 2. Bodega destino: combobox que permite la búsqueda de bodegas. 3. Artículo: combobox que permite la búsqueda de un artículo y se mostrará el stock del mismo mediante un evento Ajax. 4. Concepto: texto que permite incluir una razón del egreso. 5. Cantidad: texto donde se ingresa la cantidad requerida. 6. Agregar: Botón que permite agregar a una tabla los datos proporcionados. 7. Remover: botón que permite remover artículos de la tabla. 	



8. Procesar ingreso: botón que permite procesar toda la información lista del egreso.
<p>Observaciones: El jefe de bodega podrá especificar el concepto del ingreso, el cual debe ser obligatorio dentro del proceso, caso contrario se muestra como mensajes de alerta.</p>

Tabla 22
Historia de Usuario 16

Historia de Usuario	
Número: 16	Usuario: Jefe Bodega
Nombre historia: Control de Alertas de Stock	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media
Puntos estimados: 4.0	Iteración asignada: 1
Programador responsable: Franz Gualoto , Rubén Ortega	
<p>Descripción:</p> <p>El usuario del sistema tiene la opción de verificar Stock de Bodega mediante la opción Stock B-A, se desplegará una lista de Artículos en un ComboBox, mediante un click se desplegará los Artículos en una Tabla dinámica, el cual muestra la cantidad de stock mínimo del mismo y el usuario verificará para cada artículo y bodega que se puede satisfacer de la demanda del stock disponible en cada bodega, que será menor a 9 ítems.</p>	
<p>Observaciones: El usuario tiene opciones búsqueda:</p> <ol style="list-style-type: none"> 1. Búsqueda por Nombre de Artículo 2. Búsqueda por nombre de familia 3. Búsqueda por su mejor criterio 	

Tabla 23
Historia de Usuario 17

Historia de Usuario	
Número: 17	Usuario: Recolector
Nombre historia: Verificar Productos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 4.5	Iteración asignada: 1
Programador responsable: Franz Gualoto, Rubén Ortega	
Descripción: Se accede a la pantalla y se procede a verificar artículos el cual se vaya a vincular con cada código.	
Observaciones: El Recolector ingresará la fecha inicial y final para la búsqueda de solicitudes aprobadas por el administrador.	

Tabla 24
Historia de Usuario 18

Historia de Usuario	
Número: 18	Usuario: Recolector
Nombre historia: Asignar Códigos a Productos	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Puntos estimados: 4.5	Iteración asignada: 1
Programador responsable: Franz Gualoto , Rubén Ortega	
Descripción: El recolector asignará códigos a cada uno de los artículos y se grabará en las tarjetas RFID(Radio Frecuencia)	
Observaciones: El Recolector asignará a cada uno de los artículos con su respectivo tag.	

3.2.4 Definición de prioridades

Como prioridades fundamentales se tendrá en cuenta al momento de desarrollar el sistema, dicha característica será insertada como cualidades específicas en el diseño, con el fin de cumplir cada uno de los requerimientos de rendimiento, las cuales se muestran a continuación.

Prioridad en el negocio Alta:

1. Apariencia llamativa
2. Apariencia intuitiva
3. Apariencia amigable
4. Autenticación de Usuarios
5. Autenticación de Proveedores
6. Visualización del Stock
7. Lectura y escritura de Tags RFID(Radio Frecuencia)
8. Petición de Nuevos Artículos
9. Solicitud de Compras
10. Búsqueda de Artículos
11. Alertas de Stock Máximo y Mínimo

Prioridad en el negocio Media:

1. Búsqueda de Proveedores
2. Búsqueda de Usuarios
3. Aprobación de Solicitudes de compra
4. Informe de Stock Máximo y Mínimo
5. Carga de artículos nuevos

Prioridad en el negocio Baja:

1. Gestión dar de baja Artículos

3.2.5 Definición de iteraciones

Iteración significa el acto de repetir un proceso con el objetivo de alcanzar una meta deseada, objetivo o resultado. Cada repetición del proceso también se le denomina una “iteración”, y los resultados de una iteración se utilizan como punto de partida para la siguiente iteración.(anonimo, e-mooc, 2014)

El presente sistema de investigación se definió en tres iteraciones, cada una cuenta con su historia de usuario a ser implementadas. Es así en la primera iteración se busca integrar un primer prototipo de kardex, donde se pueda establecer los requerimientos desarrollados de manera correcta.

En la segunda iteración se tiene como objetivo fundamental buscar un alto grado de madurez del sistema donde se cumplan los requerimientos emitidos mediante cada una de las historias de usuario.

En la tercera y última iteración del sistema será únicamente para su mantenimiento, ajustes de infraestructura y de ubicación de bienes, también se evaluará posibles cambios. Esto ya se contará con un sistema completamente funcional de acuerdo a cada una de las historias de usuario como se menciona a continuación.

Primera iteración

1. Nueva apariencia
2. Guía de usuario
3. Directorio de servicios
4. Ingreso de clientes y proveedores
5. Visualización de kardex, usuarios y proveedores
6. Autenticación de roles de usuario
7. Gestión de administrador, jefe de almacén, recolector y artículos.

Segunda iteración

1. Lectura de artículos radio frecuencia.

2. Solicitud de compra de artículos.
3. Solicitud de nuevos proveedores.
4. Alertas e Informe de stock máximo y mínimo.
5. Informe de stock.
6. Gestión de inventarios.

Tercera iteración

1. Organizar físicamente cada uno de los artículos.
2. Generar informes de productos en existencia y devaluados.

3.3 Diagrama entidad relación

El modelo entidad relación está creado en base a la percepción del mundo real de un objeto en estudio, el cual está formado por entidades y relaciones entre objetos que convergen entre sí, tal como se muestra en la figura 10.

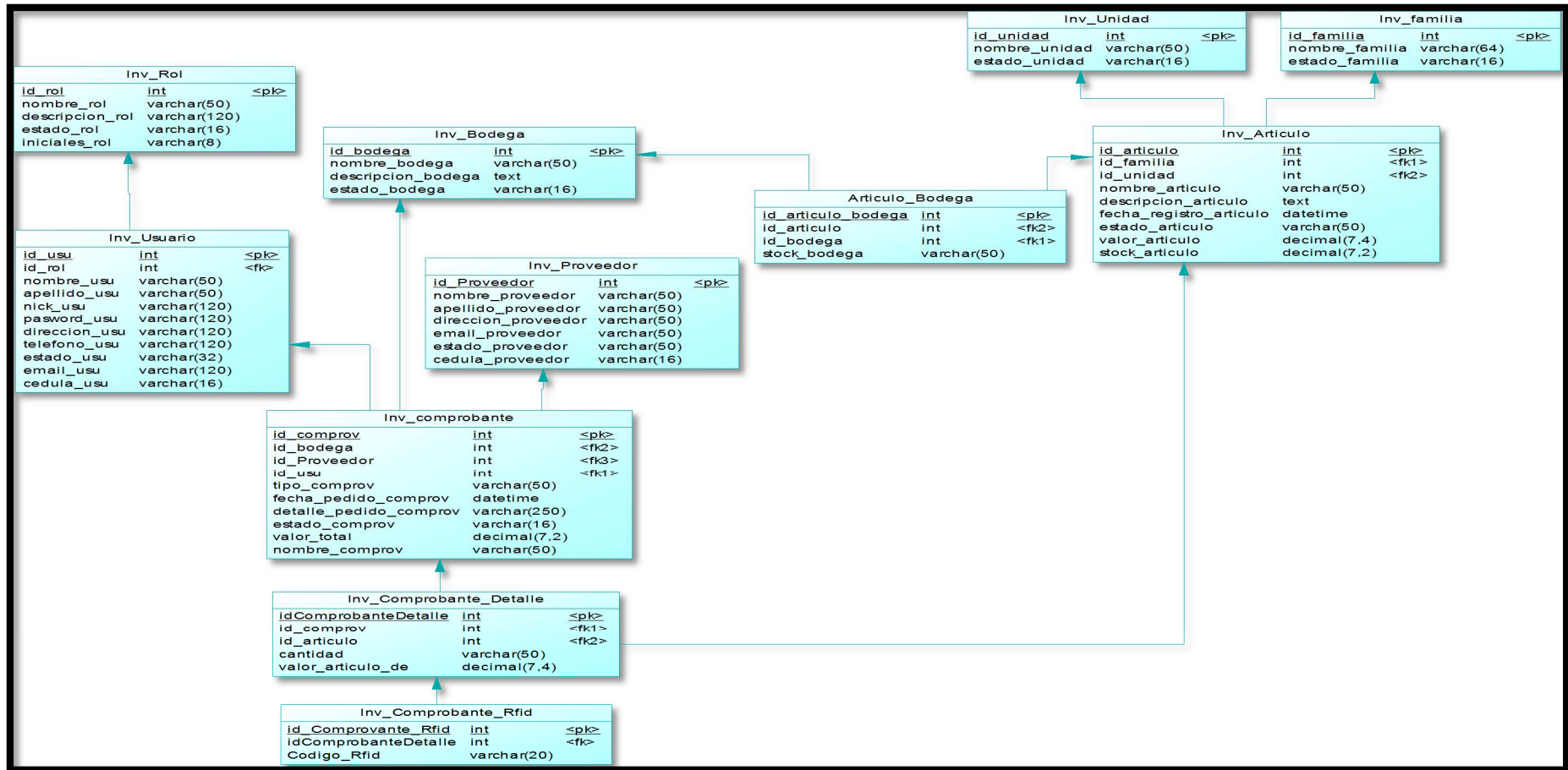


Figura 10. Diagrama Entidad Relación

3.4 Diagrama de clases

El diagrama de clases está creado para describir la estructura del sistema mediante las relaciones entre sus clases que lo constituyen, tal como se muestra en la figura 9.

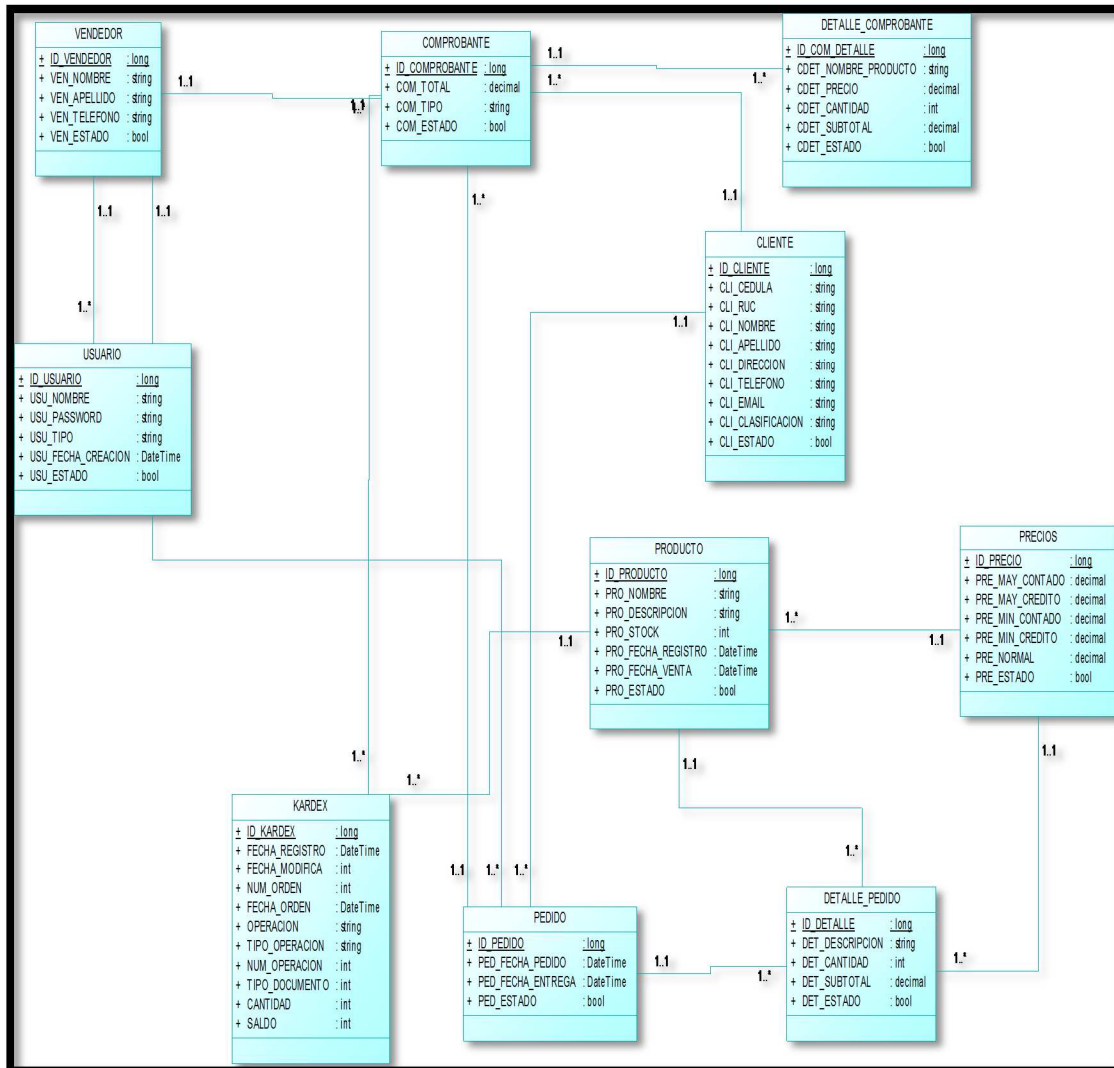


Figura 11. Diagrama de Clases

3.5 Diagrama de Despliegue

El diagrama de despliegue está creado para desplegar en su modelado cada uno de los artefactos que lo constituye como son software en nodos y sus relaciones, tal como se muestra en la **figura 11**.

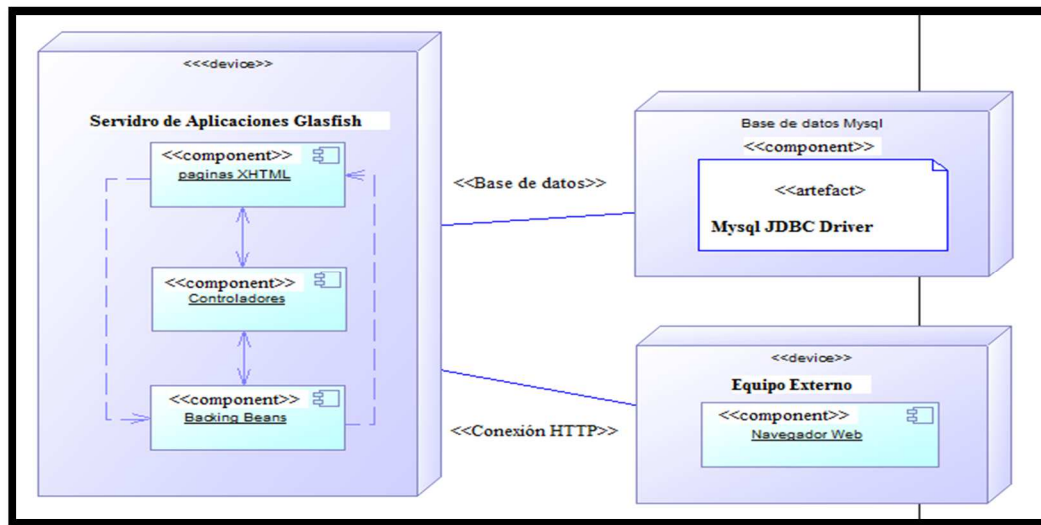


Figura 12. Diagrama de Despliegue

3.6 Diagrama de componentes

El diagrama de componentes está creado para la representación en componentes y sus respectivas dependencias entre ellos, tal como se muestra en la figura 12.

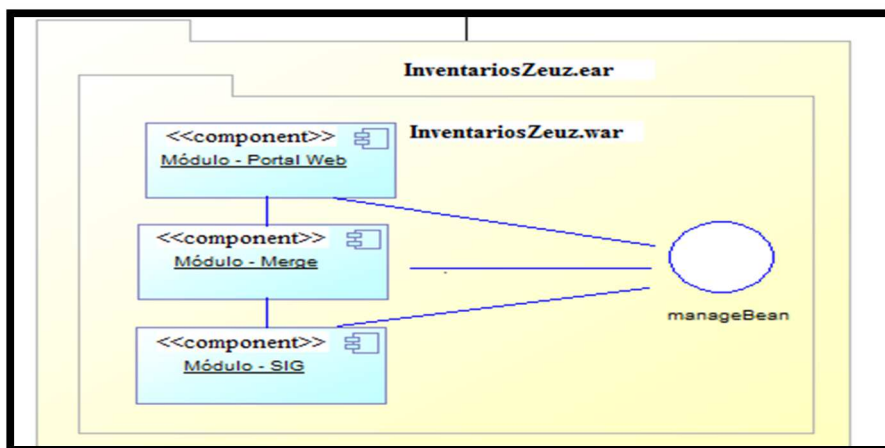


Figura 13. Diagrama de Despliegue

CAPÍTULO IV CODIFICACIÓN Y PRUEBAS

4.1 Introducción

El presente proyecto se engloba en la necesidad de negocio de gran envergadura debido a su nivel de complejidad del manejo de información, por lo cual demanda una aplicación orientada a la web que sea robusta, alta disponibilidad y amigable en cada una de sus interfaces.

El uso de nuevas tecnologías web para la plataforma Java EE requiere en el transcurso del desarrollo sea potente y su codificación se simplifique de manera deseada se ha empleado algunas nuevas tecnologías las cuales se muestran a continuación:

JavaServer Faces (JSF): JSF es una tecnología que permite el desarrollo rápido de aplicaciones y la lógica de negocio se implementa en java, o es llamada desde java, creando páginas para las vistas muy sencillas (salvo que se introduzca mucha maquetación HTML o Javascript). (Almirón, 2009)

PrimeFaces JSF 2.0: PrimeFaces es una librería de componentes visuales open source desarrollada y mantenida por Prime Technology, una compañía Turca de IT especializada en consultoría ágil, JSF, Java EE y Outsourcing. (Lerma, 2010)

Primefaces cumple con todas esas características, por lo que puede ser una buena elección. Además hay que tener en cuenta que no está limitado a utilizar una única librería. (Lerma, 2010)

JavaBeans Empresariales (Enterprise JavaBeans, EJB): Es una clase Java que tiene métodos y propiedades el cual mediante su empleo puede construir componentes que implementan la lógica de negocio de la aplicación y que

principalmente pueden ser reutilizables en cualquier parte de la vista o los otros JavaBeans dependiendo si son de vista o de control.

4.1.1 Configuración del servidor de aplicaciones GlassFish Server

4.1.2 Persistencia

La configuración de la persistencia es muy importante en el sistema ya que provee la conexión con la base de datos y puede ser utilizada en cada uno de los diferentes beans tanto de vista como de sesión.

Persistence.xml: Es un fichero muy importante el cual se encuentra ubicado en el paquete Sistema_Inventarios_Tesis-ejb dentro del paquete META-INF como se muestra en la figura 14.

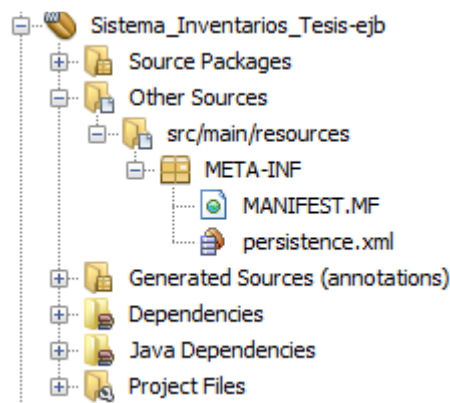


Figura 14. Configuración de la persistencia “PERSISTENCE.XML”.

Este fichero gestiona el pool de conexión generado desde el servidor de aplicaciones web de glassfish 4 e indican con que implementación JPA está trabajando y cuáles son las entidades del proyecto, como se muestra en la figura 15.

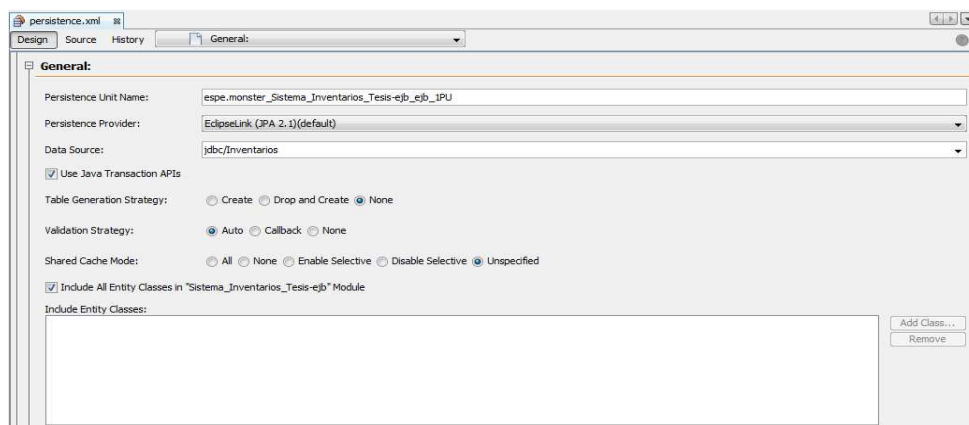


Figura 15. Unidad de persistencia

Pool de conexiones: Para administrar de mejor manera las conexiones de los clientes a la base de datos que requieran consultar o actualizar datos, se utiliza un Pool de conexiones que es controlado por un servidor de aplicaciones y permite reutilizar estas conexiones cuantas veces sean necesario.

El número de conexiones abiertas a una base de datos se limita, ya que consumen muchos recursos en el DBMS y este necesita más memoria y tiempo de procesamiento por cada conexión. La ventaja de un pool es que permite la escalabilidad y performance de una aplicación.

- a. Para crear el pool de conexiones del proyecto primero se debe inicializar el Servidor GlassFish 4, y se creará desde el IDE de Netbeans en la pestaña Services clic en Servers, clic derecho en “GlassFish Server 4” y por último clic en Start como se muestra en la figura 16.

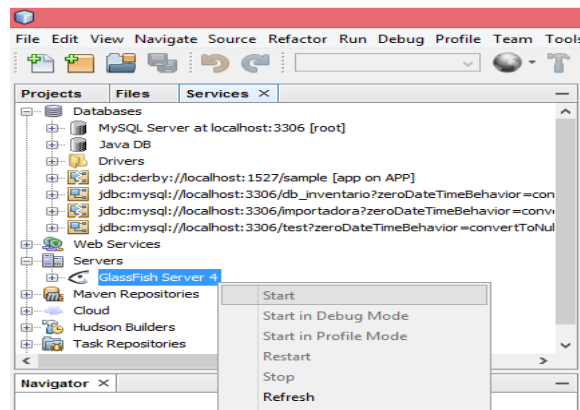


Figura 16. Iniciar Servicio GlassFish

- b. En un navegador web se podrá inicializar la página principal del servidor para administrar la conexión, se colocará como url localhost:4848 el cual nos permite ingresar a la pantalla de administración del GlassFish, como se muestra en la figura 17.

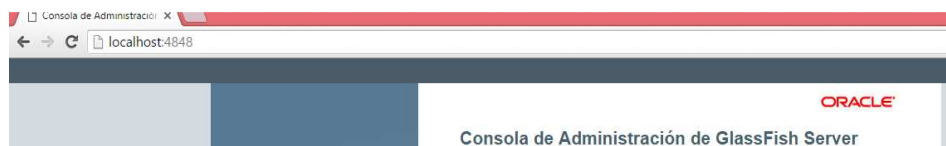


Figura 17. Administrador del GlassFish

- c. Se procede a crear el Pool de conexiones al hacer clic en JDBC >> Pools de conexiones JDBC >> Nuevo, como se muestra en la figura 18.

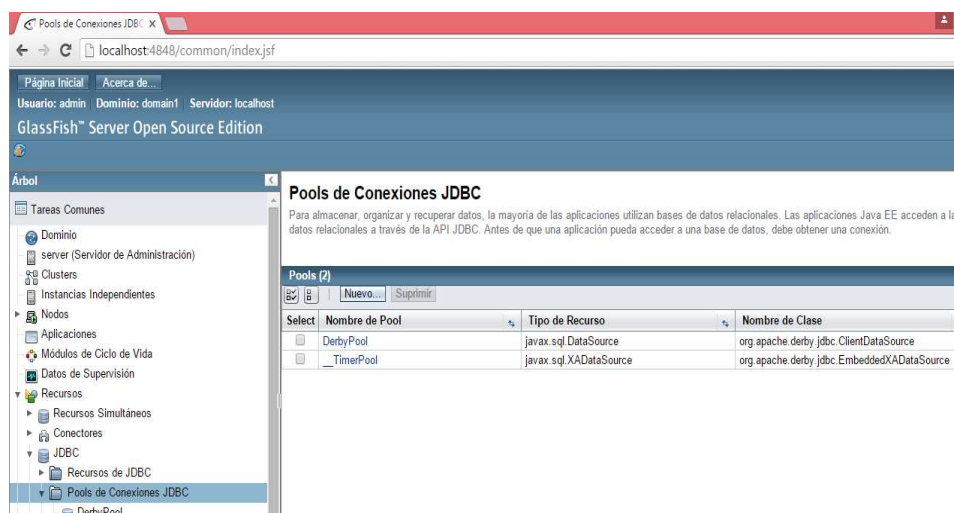


Figura 18. Menú creación Pool de Conexiones

- d. Ingresar el nombre con el cual se identifica el pool, el tipo de recurso y el proveedor de base de datos y clic en Siguiente, como se muestra en la figura 19.

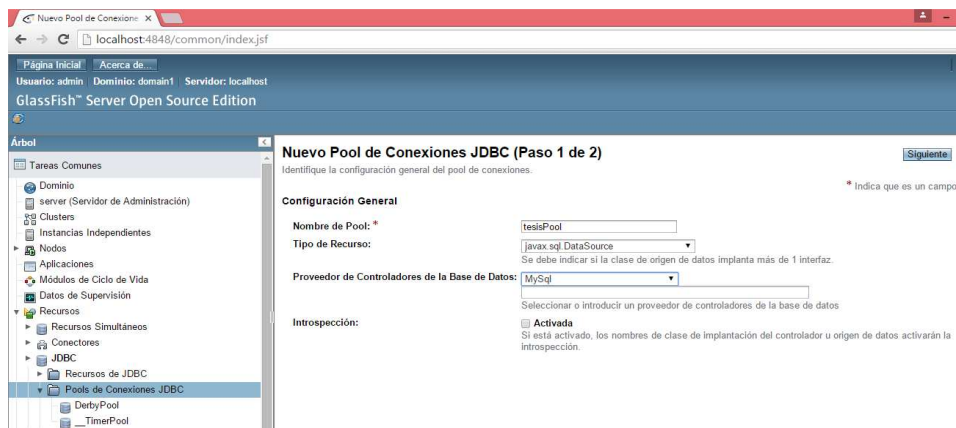


Figura 19. Ingreso parámetros del Pool de conexiones.

- e. A continuación se debe especificar los parámetros de conexión y se requiere llenar los parámetros especificados a continuación, como se muestra en la figura 19.

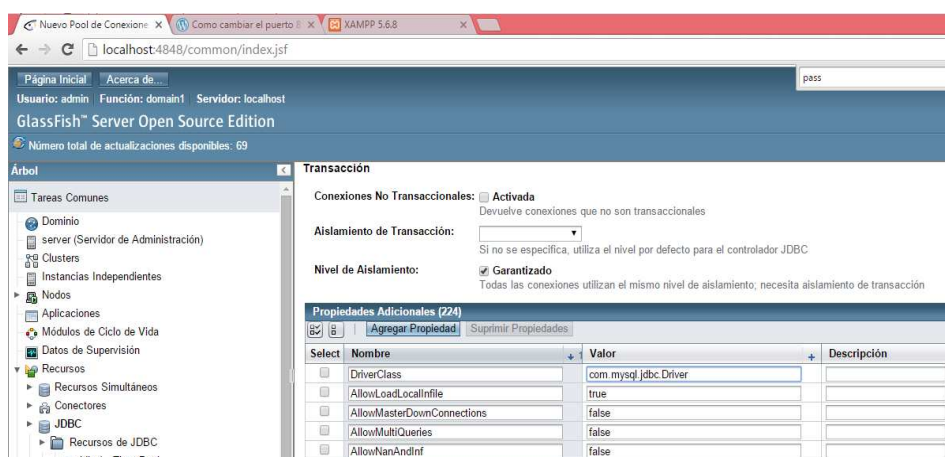


Figura 20. Ingreso de parámetros de conexión a la base de datos.

DatabaseName = <Nombre de la base de datos>

Password	= <Password de la base de datos>
Port	= <Puerto de la base de datos – 3306 para MySQL>
PortNumber	= <Puerto de la base de datos – 3306 para MySQL>
ServerName	= <Nombre del servidor de base de datos>
Url	= <Cadena conexión a la base de datos>
URL	= <Cadena conexión a la base de datos>
User	= <Nombre de usuario de la base de datos>
DriverClass	= <Nombre del controlador de la base de datos>

Para los atributos URL/Url y DriverClass los datos se pueden copiar de la conexión existente en el proyecto de Netbeans haciendo clic en Services >> Databases buscar el jdbc del proyecto en este caso jdbc:mysql://localhost:3306/db_inventario?zeroDateTimeBehavior=convertToNull [root on Default schema], hacer clic derecho y clic en propiedades, como se muestra en la figura 21.

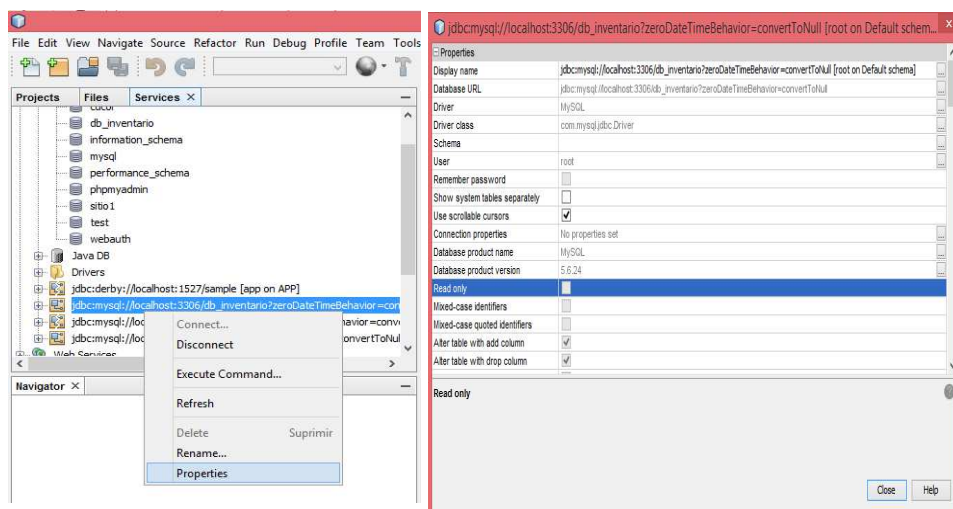


Figura 21. Obtención de Datos de conexión a la base de datos.

- f. Por ultimo Clic en finalizar.
- g. Para comprobar que el pool está bien configurado hacer clic en JDBC >> Pools de conexiones JDBC >> <Nombre del pool del proyecto> y clic en PING. A continuación debe desplegar un mensaje de “Ping realizado correctamente”. como se muestra en la figura 22.

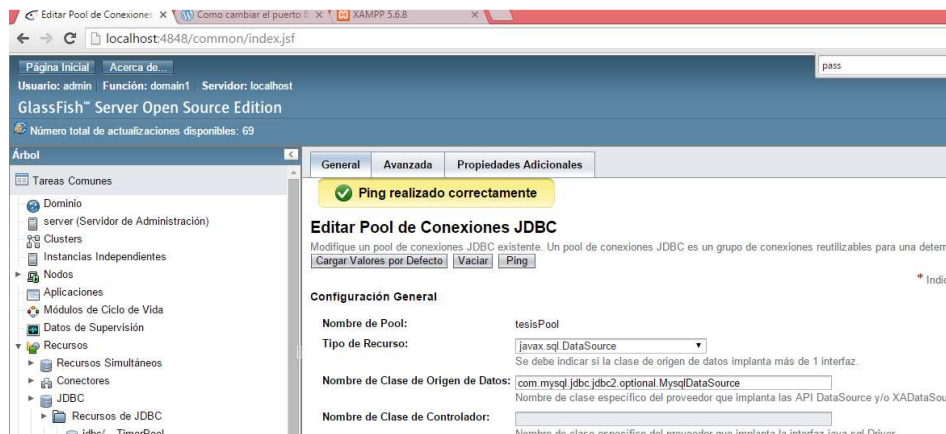


Figura 22. Comprobación de conexión del Pool.

- h. Ahora crear el enlace de JDBC hacer clic en JDBC >>Recursos de JDBC >> Nuevo, como se muestra en la figura 23.

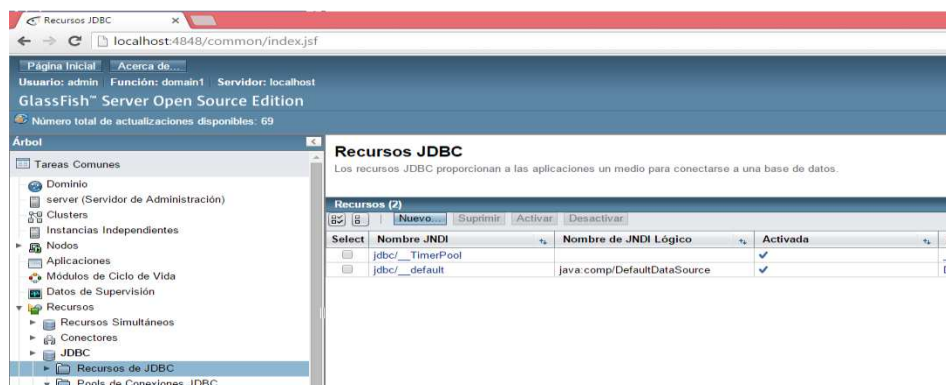


Figura 23. Creación de recursos de JDBC

- i. Ingresar los datos requeridos y clic en Aceptar

Nombre JNDI = <Nombre identificativo del recurso JDBC>

Nombre del Pool = <Nombre del Pool creado previamente>

Como se muestra en la figura 24.



Figura 24. Ingreso de parámetros del recurso de JDBC

4.1.3 Desarrollo de la aplicación web

El presente sistema se basó en el patrón Modelo Vista Controlador, el modelo está manejado por Sistema_Inventarios_Tesis-ejb, lo que corresponde a la vista y controlador maneja Sistema_Inventarios_Tesis-web como se muestra en la figura 25.

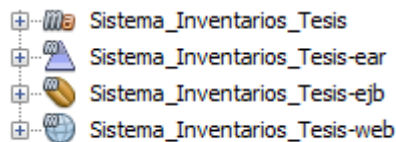


Figura 25. Paquetes del Proyecto

4.1.4 Anotaciones

El código se encuentra documentado con distintas anotaciones y clases de beans que en el proceso de la aplicación de uso.

4.1.5 Anotación en el modelo

En el modelo del sistema se tiene distintas anotaciones tal cual empiezan con el signo de @ como son Stateless que son beans sin estado, el cual permiten hacer operaciones sin estado, debido a que cada una de estas se genera automáticamente y son precisadas como clases de javax.persistence.EntityManager y javax.persistence.PersistenceContext son interfaces virtuales que se interactúan con la persistencia las cuales están definidas en las líneas de código como import como se muestra en la figura 26.

```
package espe.monster.modelo;

import espe.monster.dao.InvArticulo;
import javax.ejb.Stateless;
import javax.persistence.EntityManager;
import javax.persistence.PersistenceContext;

/**
 *
 * @author Franz
 */
@Stateless
```

Figura 26. Anotaciones del modelo.

En el modelo del sistema se tiene distintas anotaciones y empiezan con el signo de @ como son Entity, Table, XmlRootElement, NamedQueries, los mismos que permiten hacer operaciones como mapear la entidad que sea requerida, como clases de javax.persistence y javax.validation.constraints.Size, y también están definidas en las líneas de código como import como se muestra en la figura 27.

- @Entity: Se aplica a la clase e indica que ésta clase Java es una entidad a persistir. Es una anotación estándar de JPA que se puede persistir. (admin, 2012)
- @XmlRootElement: JAXB es una tecnología Java que permite mapear clases de Java a representaciones XML. JAXB provee dos funciones fundamentales: (ingmmurillo, 2010)
 - marshal. Consiste en presentar un objeto Java en XML. Java Object > XML. (ingmmurillo, 2010)

- unmarshal. Consiste en presentar XML en un objeto Java.
XML > Java Object. (ingmmurillo, 2010)
- @Table(name=""): Se aplica a la clase e indica el nombre de la tabla de la base de datos donde se persistirá la clase. Es opcional si el nombre de la clase coincide con el de la tabla. Es una anotación estándar de JPA. (admin, 2012)
- @Id: Se aplica a una propiedad Java e indica que éste atributo es la clave primaria. Es una anotación estándar de JPA. (admin, 2012)
- @Column(name=""): Se aplica a una propiedad Java e indica el nombre de la columna de la base de datos en la que se persistirá la propiedad. Es opcional si el nombre de la propiedad Java coincide con el de la columna de la base de datos. Es una anotación estándar de JPA. (admin, 2012)

```

import java.io.Serializable;
import java.util.List;
import javax.persistence.Basic;
import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.Lob;
import javax.persistence.ManyToOne;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.OneToMany;
import javax.persistence.Table;
import javax.validation.constraints.Size;
import javax.xml.bind.annotation.XmlRootElement;
import javax.xml.bind.annotation.XmlTransient;
/**
 *
 * @author Franz
 */
@Entity
@Table(name = "inv_bodega")
@XmlRootElement
@NamedQueries({
    @NamedQuery(name = "InvBodega.findAll", query = "SELECT i FROM InvBodega i"),
    @NamedQuery(name = "InvBodega.findByIdBodega", query = "SELECT i FROM InvBodega i WHERE i.idBodega = :idBodega"),
    @NamedQuery(name = "InvBodega.findByNombreBodega", query = "SELECT i FROM InvBodega i WHERE i.nombreBodega = :nombreBodega"),
    @NamedQuery(name = "InvBodega.findByEstadoBodega", query = "SELECT i FROM InvBodega i WHERE i.estadoBodega = :estadoBodega")})

```

Figura 27. Anotación estándar de JPA

La palabra reservada import en cada una de las clases de java se utiliza para "importar" o hacer referencia al o los contenidos de los paquetes en java que sean propios de la clase o se hayan creado. Entre las principales se detalla a continuación:

javax.persistence.Basic: La anotación básica se puede aplicar a una propiedad o instancia persistente variables de cualquiera de los siguientes tipos: tipos primitivos de Java, contenedores de tipo primitivo, String, java.math.BigInteger, java.math.BigDecimal, java.util.Date, java.util.Calendar, java.sql.Date, java.sql.Time, java.sql.Timestamp, byte[], Byte[], char[], Character[], enumeraciones, y cualquier otro tipo que implementa java.io.Serializable. (anonimo, Oracle, 2011)

javax.persistence.CascadeType: Define el conjunto de operaciones en cascada que se propagan a la entidad asociada. (javax.persistence)

javax.persistence.Column: Especifica la columna asignada una propiedad persistente o campo. (javax.persistence)

javax.persistence.Entity: Especifica que la clase es una entidad. (javax.persistence)

javax.persistence.GeneratedValue: Provee para la especificación de las estrategias de generación de los valores de las claves principales. (javax.persistence)

javax.persistence.GenerationType: Define los tipos de estrategias primarias de generación de claves. (javax.persistence)

javax.persistence.JoinColumn: Especifica una columna para unirse a una asociación o entidad colección de elementos. (javax.persistence)

javax.persistence.Lob: Especifica que una propiedad persistente o campo deben persistieron como un gran objeto a un tipo de objeto grande apoyado por la base de datos. (javax.persistence)

javax.persistence.ManyToOne: Especifica un solo valor asociación a otra clase de entidad que tiene muchos-a-uno multiplicidad. (javax.persistence)

javax.persistence.NamedQueries: Especifica múltiples consultas en lenguaje de consulta de Java Persistence con nombre. (javax.persistence)

javax.persistence.NamedQuery: Especifica una estática, consulta con nombre en el lenguaje de consulta de Java Persistence. (javax.persistence)

javax.persistence.OneToOne: Especifica una asociación de muchos valorada con uno-a-muchos multiplicidad. (javax.persistence)

javax.persistence.Table: Especifica la tabla principal para la entidad anotada. (javax.persistence)

4.1.6 Anotaciones en la vista

En el modelo de la vista se tiene distintas etiquetas que empiezan con la palabra reservada `xmlns` y sus distintas anotaciones como se muestra en la figura 28.

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<ui:composition xmlns="http://www.w3.org/1999/xhtml"
  xmlns:h="http://java.sun.com/jsf/html"
  xmlns:f="http://java.sun.com/jsf/core"
  xmlns:ui="http://java.sun.com/jsf/facelets"
  xmlns:p="http://primefaces.org/ui"
  template="/plantilla.xhtml">
</ui:composition>
```

Figura 28. Anotaciones en la vista.

- **Declaración DTD:** Una DTD es un documento que define la estructura de un documento XML: los elementos, atributos, entidades, notaciones, etc, que pueden aparecer, el orden y el número de veces que pueden aparecer, cuáles pueden ser hijos de cuáles, etc. El procesador XML utiliza la DTD para verificar si un documento es válido, es decir, si el documento cumple las reglas del DTD. (Marco, 2015)
- **xmlns=http://www.w3.org/1999/xhtml:** El W3C inicia la Actividad de Internacionalización en un intento por asegurar que estos formatos y

protocolos puedan utilizarse de forma universal en todos los idiomas y en todos los sistemas de escritura. (W3C España, 2010)

- **Facelets:** Los Facelets han sido creados para reemplazar a las jsp's como mecanismo para declarar vistas. Lo que se ha querido conseguir es proporcionar un modelo-vista-controlador "limpio" al no incluir código Java en el lenguaje de marca o etiqueta, Es un lenguaje ligero de declaración de páginas para la construcción de vistas JavaServer Faces usando plantillas de estilo HTML y la creación de árboles de componentes. Con las siguientes características: (Hernando, 2013)
 - Usa XHTML para la creación de páginas web. (Hernando, 2013)
 - Soporta la librería de etiquetas de facelets, además de Jsf y JSTL. (Hernando, 2013)
 - Soporta expresiones EL (Expression Language). (Hernando, 2013)
 - Plantillas para páginas y componentes. (Hernando, 2013)
 - Reutilización de código, por el uso de plantillas y la composición de componentes. (Hernando, 2013)
 - Muy rápido de compilar y renderizar, con validaciones en tiempo de compilación. (Hernando, 2013)

4.1.7 Anotaciones en el control

En el modelo de control se tiene distintas etiquetas que empiezan con la palabra reservada @ y sus distintas anotaciones como se muestra en la figura 29.

```
package servicios;
import espe.monster.modelo.InvRolFacade;
import espe.monster.dao.InvRol;
import java.util.List;
import javax.ejb.EJB;
import javax.ejb.Stateless;
import javax.ejb.LocalBean;
import javax.ejb.TransactionAttribute;
import javax.ejb.TransactionAttributeType;
/**
 *
 * @author Franz
 */
@Stateless
@LocalBean
```

```
package controles;

import espe.monster.dao.InvRol;
import java.io.Serializable;
import java.util.List;
import javax.annotation.PostConstruct;
import javax.ejb.EJB;
import javax.faces.application.FacesMessage;
import javax.faces.bean.ManagedBean;
import javax.faces.bean.SessionScoped;
import javax.faces.context.FacesContext;
import org.primefaces.context.RequestContext;
import org.primefaces.event.SelectEvent;
import servicios.RolService;

/**
 *
 * @author Franz
 */
@ManagedBean
@SessionScoped
```

Figura 29. Anotaciones de control.

CAPÍTULO V

RESULTADOS OBTENIDOS Y PRUEBAS

5.1 Introducción

Dentro de la metodología XP (Extreme Programming) incluye la fase de iteraciones la generación de pruebas de aceptación con el cliente sobre el software funcional, de ésta manera se establecerá que el desarrollo cubre en su totalidad los requerimientos y necesidades planteadas al inicio, mediante historias de usuario, lo cual se medirá y se realizará varias pruebas dentro de cada uno de los procesos para medir sus resultados y ser aprobados, ésta fase es muy importante ya que el cliente se familiarizará más a fondo con el sistema y tendrá en cuenta cada una de las secciones establecidas.

5.1.1 Pruebas de aceptación

Las pruebas de aceptación (también conocida como prueba de aceptación del usuario) es un tipo de ensayos se realicen con el fin de verificar si el producto ha sido desarrollado de acuerdo con las normas y criterios establecidos y cumplen con todos los requisitos especificados por el cliente. Este tipo de pruebas se lleva a cabo generalmente por un usuario / cliente donde se desarrolla el producto externamente por otra parte. (Computadora, 2012)

Las pruebas de aceptación fueron desarrolladas con la metodología de “Pruebas de caja negra”, el cual permite evaluar el funcionamiento global del sistema y no toma en consideración la codificación interna del mismo, y el resultado es contrarrestado con los requisitos establecidos por la empresa. Las pruebas de aceptación del usuario son consideradas como una de las pruebas más importantes ya que es el usuario las que la validan antes de que el sistema es finalmente entregado. (Computadora, 2012)

5.1.2 Matriz de Pruebas de aceptación

En la matriz de aceptación de pruebas, es una sección detallada mediante una tabla del sistema a ser probado, teniendo en cuenta como previa información como son los requisitos, instrucciones o resultados esperados, estos ayudarán al cliente a enfocarse como está desarrollado el software en cada una de estas pruebas y así tener en cuenta claro el funcionamiento de cada prueba.

En cada prueba se tendrá registrado los datos de usuario que interactúa en cada una de las matrices como son la fecha de ejecución, responsable de la prueba, lugar donde se ejecutó y su estado, también se tendrá en cuenta las garantías de cada prueba con las que fueron realizadas principalmente como se muestran en las tablas N 19,20,21,22,23,24,25,26,27,28,29

Tabla 25
Prueba de aceptación 1

Prueba de Aceptación	
ID	P001
Descripción	Apariencia del sistema de inventarios Zeuz Sistemas
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	4. Ingrese a la siguiente url desde un navegador, localhost:8080/Sistema_Inventarios_Tesis-web
Resultados Esperador	Se muestra en el navegador la interfaz de bienvenida del sistema de inventarios Zeuz sistemas.
Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváez
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

Tabla 26
Prueba de aceptación 2

Prueba de Aceptación	
ID	P002
Descripción	Mantenimiento de Rol de Usuarios
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	<ol style="list-style-type: none"> 1. Ingrese a la parte de mantenimientos de sistema, y colocarse en la parte de Rol. 2. Ingresa un nuevo rol de usuario.
Resultados Esperador	Se muestra en la pantalla la creación de nuevo rol.
Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváz
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

Tabla 27
Prueba de aceptación 3


Prueba de Aceptación	
ID	P003
Descripción	Mantenimiento de Usuarios
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	<ol style="list-style-type: none"> 1. Ingrese a la parte de mantenimientos de sistema, y colocarse en la parte de Usuario. 2. Ingresa un nuevo rol de usuario.
Resultados Esperador	Se muestra en la pantalla la creación de nuevo Usuario.
Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváz
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

Tabla 28
Prueba de aceptación 4

Prueba de Aceptación	
ID	P004
Descripción	Mantenimiento de Proveedor
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	<ol style="list-style-type: none"> 1. Ingrese a la parte de mantenimientos de sistema, y colocarse en la parte de Proveedor 2. Ingresa un nuevo rol de usuario.
Resultados Esperador	Se muestra en la pantalla la creación de nuevo Proveedor.
Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváez
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

Tabla 29
Prueba de aceptación 5

Prueba de Aceptación	
ID	P005
Descripción	Mantenimiento de Bodega
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	<ol style="list-style-type: none"> 1. Ingrese a la parte de mantenimientos de sistema, y colocarse en la parte de Bodega 2. Ingresa un nuevo rol de usuario.
Resultados Esperador	Se muestra en la pantalla la creación de nuevo Proveedor.

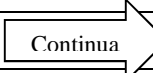
Continúa 

Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváez
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

Tabla 30
Prueba de aceptación 6

Prueba de Aceptación	
ID	P006
Descripción	Mantenimiento de Artículo
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	<ol style="list-style-type: none"> 1. Ingrese a la parte de mantenimientos de sistema, y colocarse en la parte de Artículo. 2. Ingresa un nuevo rol de usuario.
Resultados Esperador	Se muestra en la pantalla la creación de nuevo Proveedor.
Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváez
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

Tabla 31
Prueba de aceptación 7

Prueba de Aceptación	
ID	P007
Descripción	Mantenimiento de Unidad
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	

	<ol style="list-style-type: none"> 1. Ingrese a la parte de mantenimientos de sistema, y colocarse en la parte de unidad. 2. Ingresa un nuevo rol de usuario.
Resultados Esperador	Se muestra en la pantalla la creación de nuevo Proveedor.
Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváez
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

Tabla 32
Prueba de aceptación 8

Prueba de Aceptación	
ID	P008
Descripción	Mantenimiento de Familia
Inicialización	Asegurar que el servidor de base de datos y el servidor de la aplicación también estén activos.
Instrucciones	<ol style="list-style-type: none"> 1. Ingrese a la parte de mantenimientos de sistema, y colocarse en la parte de Familia. 2. Ingresa un nuevo rol de usuario.
Resultados Esperador	Se muestra en la pantalla la creación de nuevo Proveedor.
Fecha de ejecución	Miércoles, 10 de febrero del 2016
Ejecutado por	Ing. Guillermo Narváez
Lugar de ejecución	Oficina Zeuz Sistemas
Estado de Prueba	Concluida

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1 Introducción

En la culminación del sistema de inventarios refleja que se ha cumplido con cada uno de los objetivos y alcances planteados, a continuación se presentan las conclusiones y recomendaciones.

6.1.1 Conclusiones

- Para llevar a cabo el presente proyecto de desarrollo de software se utilizó la metodologías XP (Extreme programmind) durante todo el ciclo de vida del proyecto, el cual permitió obtener un producto acorde con las necesidades planteadas, afrontando eficientemente los cambios inesperados sobre los requerimientos y dentro de los plazos previstos.
- El resultado de las pruebas realizadas en el inventario para diferentes cantidades de ítems y comparando con la tecnología barcode que estaba implementada en la empresa “Zeuz Sistem” arrojó resultados positivos para Rfid ya que la velocidad de lectura de la nueva tecnología implementada supero en un 78,44% a el código de barras, permitiendo optimizar tiempo y recurso a la empresa.
- Otra ventaja es que la nueva tecnología nos permite muy independiente de contener un código único el almacenamiento de datos en su chip interno, dándole mayor flexibilidad y reusabilidad, también cuenta con dos claves internas que hacen al sistema seguro que no se pueda falsificar ni duplicar datos, algo que en el código de barras es imposible de hacer. También el rendimiento y durabilidad de Rfid
- Es importante tomar en cuenta que los costos de implementación y mantenimiento de la tecnología Rfid es más elevada que el uso de código de barras, pero este costo se ve opacado por los grandes beneficios que brinda Rfid al optimizar los recursos y el tiempo de los procesos en la empresa.

- Este proyecto permitirá a la empresa “Zeuz System” tener un inventario y control de stock más eficiente y eficaz, podrán saber exactamente donde se encuentra sus productos ya que el sistema permite la trazabilidad y geolocalización, otra ventaja es que con la instalación de antenas recolectoras de Rfid podrán hacer lecturas en lotes reduciendo en gran medida los tiempos de inventarios o traslados de productos hacia otras bodegas.

6.1.2 Recomendaciones

- Se recomienda el uso de este sistema ya que puede servir para el 90% de empresas que manejen artículos en bodegas diferentes y así controlar el stock de la mercadería, de manera adecuada y precisa.
- Se recomienda la plataforma de desarrollo web java, y como librería de desarrollo JSF Primefaces ya que permite desarrollar aplicaciones mucho más robustas y reutilizables dentro del mismo entorno.
- Para el correcto funcionamiento del sistema de inventarios Zion Sistemas web se recomienda estar siempre registrado como usuario de bodega para así monitorear el stock de bodega, y tener en cuenta que los tags no se encuentren dentro de algún objeto como billeteras, carteras o maletines ya que esto puede reducir el radio de cobertura entre la lectura y la etiqueta RFID,
- El uso de la tecnología RFID es muy eficaz teniendo en cuenta que las tarjetas electrónicas o tags no se encuentren en algún lugar visible para así evitar que alguien manipule o sea desprendido del sitio correcto y evitar problemas con el monitoreo de la información de cada artículo.

GLOSARIO DE TÉRMINOS.

Xp.- Programación extrema, es una metodología ágil para el desarrollo de software.

RFID.- Identificación por radio frecuencia.

Tag.- Dispositivo que contiene una antena para recibir y responder a peticiones de radio frecuencia.

Lector o recolectores RFID.- Dispositivos que se encargan de alimentar las antenas de los tags a través de las antenas y al mismo tiempo capturan sus datos.

Java.- Lenguaje de programación orientada a objetos, desarrollado por Sun Microsystem.

Mysql.- Sistema de gestión de base de datos relacional, multi-hilo y multiusuario.

Inventario.- Registro de bienes u objetos pertenecientes a una persona física.

Kardex.- Documento o registro administrativo de control, que incluye datos generales del bien u objeto.

Stock.- Conjunto de mercaderías o productos, almacenados para la venta o comercialización.

ERS.- Especificación de requerimientos de software.

Requerimientos.- Características que debe tener un software para poder soportar o ejecutar una aplicación en un dispositivo específico.

Primefaces.- Librería de componentes enriquecidos para JavaServer Faces (JSF), que facilitan la creación de aplicaciones web.

REFERENCIAS BIBLIOGRAFICAS

- © 2014 Anirban Adak. All Rights Reserved. (2014, 04 25). *Introduction to middleware RFID*. Recuperado el 06 26, 2015, de <http://www.anirbanadak.com/introduction-rfid-middleware-2>
- Acevedo, V. J. (2004, Noviembre 12). *Javeriana*. Recuperado el Noviembre 25, 2015, de Javeriana: <http://www.javeriana.edu.co/biblos/tesis/ingenieria/tesis10.pdf>
- admin. (2012, 09 16). *Aprendiendo Hibernate*. Recuperado el diciembre 23, 2015, de Aprendiendo Hibernate: http://cursohibernate.es/doku.php?id=unidades:02_hibernate:02_entidad
- Almirón, C. G. (2009, marzo 26). *AdictosAlTrabajo.com*. Recuperado el diciembre 22, 2015, de AdictosAlTrabajo.com: <http://www.adictosaltrabajo.com/tutoriales/introduccion-jsf-java/>
- anonimo. (2011, febrero 10). *Oracle*. Recuperado el diciembre 23, 2015, de Oracle: <https://docs.oracle.com/javaee/6/api/javax/persistence/Basic.html>
- anonimo. (2014, julio 12). *e-mooc*. Recuperado el noviembre 22, 2015, de e-mooc: <http://www.e-mooc.com/lecciones-en-texto/iterar-o-pivotar/>
- Benchmark, C. (2015, 06). *Identificación por radiofrecuencia (RFID)*. Recuperado el 06 26, 2015, de <http://es.ccm.net/contents/619-identificacion-por-radiofrecuencia-rfid>
- Bolivariana, U. U. (2015). *Ingeniería de software*. Recuperado el 06 26, 2015, de Programación extrema XP: http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html
- Buhoos.com. (2015). *Lenguajes de Programación cuadro comparativo*. Recuperado el 11 28, 2015, de <http://blog.buhoos.com/lenguajes-de-programacion-cuadro-comparativo/>
- Computadora. (2012, 08 29). *quecomputadoracomprar*. Recuperado el 02 09, 2016, de quecomputadoracomprar: <http://quecomputadoracomprar.com/software-testing-pruebas-de-aceptacion-2/>
- CORREA ESPINAL, A., ÁLVAREZ LÓPEZ, C., & GÓMEZ MONTOYA, Á. (2010, 07 23). *SISTEMAS DE IDENTIFICACIÓN POR RADIOFRECUENCIA, CÓDIGO DE BARRAS Y SU RELACIÓN CON LA CADENA DE SUMINISTRO*. Colombia., Bogotá.
- Electronics, K. (2010, 3 23). *kimaldi*. Recuperado el 06 28, 2015, de kimaldi: http://www.kimaldi.com/area_de_conocimiento/rfid/frecuencias_de_funcionamiento
- Hernan, K. (2012, marzo 02). *slideshare*. Recuperado el 11 28, 2015, de slideshare: <http://es.slideshare.net/KozmoHernan1/c-comparativo-servidores-servicios>
- Hernando, C. G. (2013, enero 28). *No Todo Código*. Recuperado el Diciembre 27, 2015, de No Todo Código: <http://www.notodocodigo.com/introduccion-a-jsf/facelets/>
- ingmmurillo. (2010, agosto 11). *Mis Experiencias en el Desarrollo de Software*. Recuperado el diciembre 23, 23, de Mis Experiencias en el Desarrollo de Software: <http://ingmmurillo.blogspot.com/2010/08/utilizacion-de-jaxb-paramapear-clases.html>

- javax.persistence, P. (s.f.). *Oracle*. Recuperado el diciembre 23, 2015, de Oracle: <https://docs.oracle.com/javaee/7/api/javax/persistence/package-summary.html>
- Lerma, E. V. (2010, junio 30). *adictosaltrabajo.com*. Recuperado el diciembre 22, 2015, de adictosaltrabajo.com: <http://www.adictosaltrabajo.com/tutoriales/introduccion-primefaces/>
- M^a Victoria Bueno Delgado, Alejandro S. Martínez Sala, Esteban Egea López, Javier Vales Alonso, Joan García Haro. (s.f.). *Sistemas globales de localización y trazabilidad mediante identificación. Sistemas globales de localización y trazabilidad mediante identificación*. Cartagena, Colombia.
- Marco, B. S. (2015, abril 17). *mclibre*. Recuperado el diciembre 27, 2015, de mclibre: http://www.mclibre.org/consultar/xml/lecciones/xml_dtd.html
- Mora, M. (2010, diciembre 10). *Scrib*. Recuperado el noviembre 22, 2015, de Scrib: <http://es.slideshare.net/MiquelMora/historias-de-usuario>
- München, L. –L.-M.-U. (2014, 08 27). *UWE – UML-based Web Engineering*. Recuperado el 01 15, 2015, de <http://uwe.pst.ifi.lmu.de/aboutUwe.html>
- OFBiz, O. f. (2014, 09 28). *Ciclo de vida de un proyecto XP*. Recuperado el 06 26, 2015, de <http://oness.sourceforge.net/proyecto/html/ch05s02.html>
- Pérez, L. B. (2009, 11 12). Recuperado el 06 28, 2015, de <http://itzamna.bnct.ipn.mx/dspace/bitstream/123456789/9053/1/METDESRFID.pdf>
- Rodas, D. J. (2009, 03 22). *espol*. Recuperado el 06 28, 2015, de espol: <https://www.dspace.espol.edu.ec/bitstream/123456789/6784/1/D-39214.pdf>
- Roncancio M, M., Cuevas , J., Rodríguez, J., Villalba , J., & Aguirre , C. (2011, mayo 15). *La Kardex*. Obtenido de La Kardex: <http://mugetsu-tribus-urbanas.blogspot.com/2011/05/el-kardex-y-su-importancia-en-los.html>
- S.A., P. F. (2015). *tutorial: tecnología RFID, identificación automática por radiofrecuencia*. Recuperado el 06 27, 2015, de <http://www.puntoflotante.net/RFIDTUT.htm>
- S.L, T. I. (2011, 10 11). *Tag Ingenieros RFID*. Recuperado el 6 28, 2015, de Tag Ingenieros RFID: <http://www.tagingenieros.com/?q=node/107>
- SA. (2011, 11 12). Recuperado el 6 20, 2015, de http://ice.uaz.edu.mx/c/document_library/get_file?uuid=26772c33-70b6-4983-836b-58843a21886c&groupId=54327
- Seguros, L. R. (s.f.). *Tecnología RFID (Parte III): Funcionamiento*. Recuperado el 06 18, 2015, de Tecnología RFID (Parte III): Funcionamiento: <http://www.llerandi.com/BlogDeSeguridad/tecnologia-rfid-parte-iii-funcionamiento/>
- SISTEMAS, Z. (2012, JUNIO 23). *ZION SISTEMAS*. Recuperado el 1 JULIO, 2015, de ZION SISTEMAS: <http://zionsistemas.com/>
- Tangent LLC , TES Global Limited. (2015). *Procesos del software*. Recuperado el 06 26, 2015, de Metodología XP: <http://procesosdesoftware.wikispaces.com/METODOLOGIA+XP>
- Tecnología, I. -I. (2012, 10 19). *RFID: El presente y futuro de la identificación*. Recuperado el 06 27, 2015, de <http://www.iat.es/2012/10/rfid-el-presente-y-futuro-de-de-la-identificacion/>
- Víctor Acevedo, Alejandro García, Juan Sandino. (2004, noviembre). *Sistema de registro y control de salida de elementos mediante dispositivos rfid*.

Recuperado el noviembre 26, 2015, de
<http://www.javeriana.edu.co/biblos/tesis/ingenieria/tesis10.pdf>
W3C España. (2010, julio 14). Recuperado el febrero 09, 2016, de W3C España:
<http://www.w3c.es/Divulgacion/GuiasBreves/internacionalizacion>