

## **CERTIFICACIÓN DEL DIRECTOR**

Ing. Marco Augusto Ocampo Carpio. Mg. Sc.

**DOCENTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS DE LA  
UNIVERSIDAD NACIONAL DE LOJA.**

### **C E R T I F I C A:**

Que la presente tesis con el tema “**SISTEMA WEB DE GESTIÓN DE FARMACIA, BASADO EN TECNOLOGÍA MÓVIL PARA EL HOSPITAL BÁSICO 7 B.I "LOJA"**”, previa a la obtención del título de Ingeniero en Sistemas, fue prolijamente dirigido supervisado y revisado en cada uno de sus aspectos; razón por la cual doy validez y justificación, autorizado su presentación, sustentación y publicación.



**Ing. Marco Augusto Ocampo Carpio. Mg. Sc**

**DIRECTOR**

## **AUTORÍA**

Yo, Angel Alberto Valdez Masache, declaro ser el autor del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional, la publicación de mi tesis en el Repositorio Institucional- Biblioteca Virtual.

**Autor:** Angel Alberto Valdez Masache.

**Firma:** 

**Cédula:** 1103786990

**Fecha:** 28 de enero de 2015

## **CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN DEL TEXTO COMPLETO.**

Yo, **ANGEL ALBERTO VALDEZ MASACHE**, declaro ser autor de la tesis titulada: **“SISTEMA WEB DE GESTIÓN DE FARMACIA, BASADO EN TECNOLOGÍA MÓVIL PARA EL HOSPITAL BÁSICO 7 B.I. LOJA”**, como requisito para optar al grado de: **INGENIERO EN SISTEMAS**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, veinte y ocho días del mes de enero del dos mil quince.

**Firma:**



**Autor:** Angel Alberto Valdez Masache

**Cédula:** 1103786990

**Dirección:** Clodoveo Jaramillo (Calles: Iquitos 24-78 entre Córdova y la plata)

**Correo electrónico:** betitos2309@gmail.com

**Teléfono:** 2613848 **Celular:** 0980754923

### **DATOS COMPLEMENTARIOS**

**Director de tesis:** Ing. Marco Augusto Ocampo Carpio. Mg. Sc

**Tribunal de grado:** Ing. Waldemar Victoriano Espinoza Tituana. Mg. Sc.

Ing. Mario Andrés Palma Jaramillo. Mg. Sc.

Ing. Lorena Elizabeth Conde Zhingre. Mg. Sc.

## **AGRADECIMIENTO**

Quiero expresar mis más sinceros agradecimientos a la Universidad Nacional de Loja, a la Carrera de Ingeniería en Sistemas y en especial a todos los docentes que durante mi tiempo de estudio impartieron sus conocimientos para formarme ahora en un profesional, Al ing. Marco Ocampo, por su tiempo estimado durante este periodo de desarrollo del proyecto,

Agradecer a Dios por permitirme crecer cada día más, a mis padres Angelito y Rosita, que han estado pendientes de mi esfuerzo, a mi hija Valentina, mi prometida Flor María, mis hermanos Karla, Judith y Diego, a mis amigos Lenin y Fabricio, que siempre me apoyaron y sobre todo me dieron ánimos para seguir adelante y poder culminar esta etapa de estudios.

**Angel Alberto Valdez Masache**

## **DEDICATORIA**

Dedico este logro a mis padres Angel Alberto Valdez y Rosita Fidelina Masache Quezada, que fueron el apoyo fundamental desde que inicie mis estudios universitarios y que siempre han estado al pendiente de mis estudios, que con sus consejos me han sabido dar una vida digna de hijo, y ahora yo como profesional poder devolver con esfuerzo y profesionalismo todo lo impartido por ellos.

A mi hija Valentina y a mi prometida Flor María, por ser la razón de culminar con tenacidad y esfuerzo este logro, a mis hermanos Karla, Judith y Diego que con su cariño y respaldo han hecho que siga creciendo como persona y ahora profesionalmente, también quiero dedicar este triunfo a mis cuñados, sobrinos, familia y amigos en general ya que de una u otra manera siempre estuvieron presentes apoyándome para culminar con éxito mis estudios.

Y sobre todo a Dios por ser el que siempre estuvo en mis oraciones y bendiciones que pedí, para lograr este triunfo en mi vida.

**Angel Alberto Valdez Masache**

## **CESIÓN DE DERECHOS**

La presente tesis queda a disposición de la Universidad Nacional de Loja y del Hospital Básico 7 B.I Loja, para que puedan determinar su utilización en los fines que se considera pertinente y para lo que se la desarrolló.

**Angel Alberto Valdez Masache**

**1103786990**

## **a. Título**

“SISTEMA WEB DE GESTIÓN DE FARMACIA, BASADO EN TECNOLOGÍA MÓVIL  
PARA EL HOSPITAL BÁSICO 7 B.I LOJA”

## **b. Resumen**

La presente tesis “**Sistema Web de gestión de farmacia, basado en tecnología móvil para el Hospital Básico 7 B.I. Loja**”, tiene como finalidad que el hospital y en este caso el departamento de farmacia cuente con un sistema que cumpla las necesidades específicas de las labores que se realizan en el día a día, brindando a los usuarios rapidez y mejor servicio a la sociedad.

La metodología de desarrollo aplicada a este software es la XP, la cual empieza con sesiones diarias durante semanas con el cliente, en las que se da campo abierto y claridad a los requerimientos puntuales y necesarios para el desarrollo del proyecto, esto mediante un programa ya en uso que no cumple las necesidades específicas.

Para el desarrollo del sistema se utilizó el lenguaje de programación Ruby, con la ayuda del framework Ruby on Rails el cual es código libre y brinda libertad al desarrollar sitios web, permitiendo hacer uso de herramientas como: Javascript, Css, JQuery, Sublinetext, Sourcetree, Github, Digital ocean, HTML, los mismos que fueron base fundamental para el front del sistema de farmacia. Bootstrap un framework que ayudó a que el sistema se adapte a dispositivos móviles y así realizar y lograr el objetivo planteado, además se efectuaron las pruebas de funcionalidad, seguridad, aceptación y pruebas de carga y estrés.

El sistema de gestión de farmacia se desarrolló tomando en cuenta la información que ya brindaba el sistema antes usado, permitiendo recolectar procesos y así poder implementar la nueva forma de trabajo de una forma rápida y eficiente. Cabe indicar que se realizó reportes diarios y mensuales los cuales son de gran ayuda a este departamento de farmacia en el caso de auditorías o de cualquier inspección por parte de los superiores de esta institución.

El sistema hoy en día está integrado con las dependencias de estadística y central de enfermería que se desarrolló por parte de otro programador, y así llegar a lo que se pensaba solucionar; como es el tiempo de demora de comunicación entre estas dependencias dando así mayor rapidez en el servicio hacia el cliente.



## **Summary**

This research related to the Web Pharmacy Management System is based on creating a standard technology for health care providers and services. It was designed specifically for allowing the pharmacy to better enhance its effectiveness and allows users to meet the daily objectives of customer care and pharmaceutical efficiency in a basic hospital and nursing health care environment.

The development, methodology and implementation of this system were applied to Windows XP and have been designed specifically for use in the basic hospital or mobile health care setting. It allows for an interactive program that gives instruction and feedback to users as well as direction on proper and appropriate usage.

RUBY programming language was used in the development of this software system. Additionally, framework on the RAILS was utilized which allows and gives freedom to develop and expand this particular form of information technology. This system is easily integrated into Javascript, CSS, JQUERY, Sublimetext, Sourcetree, Github, Digital Ocean, Bootstrap and HTML which are the foundational bases of current pharmacy technologies in Ecuador. Bootstrap is a technology framework which allows for the adaptation and utilization of mobile devices so that user functionality can be achieved in any health care setting.

In order to accommodate this research, the current pharmacy management system in use was studied in order to determine a solution that best meets the needs of consumers and pharmacy users. A daily report on its functionality ensures that the system continuously maintains integrity, allows for corrections and additions when needed, and gives superior customer service throughout its use.

## ÍNDICE DE CONTENIDOS

CERTIFICACIÓN DEL DIRECTOR .....	I
AUTORÍA .....	II
CARTA DE AUTORIZACIÓN DE TESIS.....	III
AGRADECIMIENTO .....	III
DEDICATORIA .....	V
CESIÓN DE DERECHOS .....	VI
a. Título .....	VII
b. Resumen.....	VIII
Summary .....	IX
ÍNDICE DE CONTENIDOS .....	X
ÍNDICE DE FIGURAS .....	XIII
ÍNDICE DE TABLAS .....	XIV
GOSARIO .....	XVII
c. Introducción .....	1
d. Revisión de Literatura. ....	3
Capítulo 1 .....	3
1. Ruby on Rails .....	3
1.1. ¿Qué es Ruby?.....	3
1.2. Introducción. ....	3
1.3. Características de Ruby.....	4
1.4. Ruby on Rails como herramienta de desarrollo de aplicaciones web...5	
1.5. Desarrollo de Aplicaciones .....	9
Capítulo 2.....	11
2. Metodología de desarrollo XP. ....	11
2.1. Programación Extrema (Extreme Programming, XP) .....	11
2.2. Roles XP.....	12
2.3. Principios Básicos.....	13

2.4. Fases de la Metodología.....	17
2.4.1. FASE 1: Planificación del Proyecto.....	18
2.4.2. FASE 2: Diseño. ....	21
2.4.3. FASE 3: Desarrollo. ....	22
2.4.4. FASE 4: Pruebas. ....	22
<b>Capítulo 3.....</b>	<b>22</b>
<b>3. Herramientas de desarrollo.....</b>	<b>22</b>
3.1. JAVASCRIPT.....	22
3.2. CSS (Hojas de estilo en cascada) .....	23
3.3. JQUERY .....	25
3.4. BOOTSTRAP.....	25
3.5. COFFEESCRIPT.....	28
3.6. MYSQL.....	29
<b>e. Materiales y Métodos.....</b>	<b>32</b>
1. Métodos.....	32
2. Técnicas.....	33
3. Metodología de desarrollo de software .....	34
<b>f. Resultados .....</b>	<b>37</b>
<b>1. FASE 1. -Planificación del Proyecto .....</b>	<b>37</b>
1.1 Introducción .....	37
1.2 Propósito.....	37
1.3 Alcance .....	37
1.4 Definiciones, siglas y abreviaciones. ....	40
2. Descripción global .....	41
2.1. Perspectiva del producto .....	41
2.2. Funciones del producto.....	41
2.3. Características del Usuario.....	44
2.4. Restricciones .....	44
2.5. Atención y dependencias.....	44
2.6. Prorratear los requisitos.....	44
3. Los requisitos específicos.....	45
3.1. Requerimientos Funcionales .....	45

3.2. Requerimientos No Funcionales.....	47
3.3.Equipo: Integrantes y Roles .....	47
3.4. Historias de Usuarios .....	48
3.5.Resumen de Historias de Usuario .....	64
3.6. Plan de Entregas .....	65
3.7. División en iteraciones.....	66
3.8. Velocidad del proyecto.....	67
3.9. Entregas Pequeñas .....	67
<b>2. FASE 2.- Diseño.....</b>	<b>67</b>
2.1. Diseños Simples .....	68
2.2. Glosario de Términos.....	68
2.3. Tarjetas CRC (Clase - Responsabilidad – Colaborador) .....	69
2.4. Modelo del Dominio .....	72
2.5. Modelo de Clase.....	73
2.6. Modelo de la Base de Datos.....	74
<b>3. FASE 3.- Desarrollo .....</b>	<b>75</b>
3.1. Cliente Siempre Presente .....	75
3.2. Pruebas Unitarias .....	75
3.3. Programación por parejas.....	77
3.4. Integración .....	77
<b>4. Fase 4.- Pruebas .....</b>	<b>77</b>
4.1. Implantación .....	77
4.2. Pruebas de Aceptación.....	77
4.3. Plan de Validación .....	93
<b>g. Discusión.....</b>	<b>107</b>
<b>h. Conclusiones. ....</b>	<b>116</b>
<b>i.Recomendaciones .....</b>	<b>117</b>
<b>j. Bibliografía .....</b>	<b>118</b>
<b>k. Anexos .....</b>	<b>120</b>

## ÍNDICE DE FIGURAS

FIGURA 1. ARQUITECTURA DE UNA APLICACIÓN DESARROLLADA CON RoR .....	6
FIGURA 2. MODELO, VISTA, CONTROLADOR EN RoR .....	9
FIGURA 3. FASES DE LA METODOLOGÍA XP (APLICADA) .....	18
FIGURA 4. CÓDIGO PARA CARGAR LAS LIBRERÍAS EN JAVASCRIPT .....	23
FIGURA 5. CÓDIGO CSS .....	24
FIGURA 6. DATATABLE NORMAL .....	24
FIGURA 7. DATATABLE APLICADO ESTILO .....	24
FIGURA 8. TAMAÑO COMPLETO EN UNA LAPTOP. ....	26
FIGURA 9. TAMAÑO DESDE UN DISPOSITIVO MÓVIL.....	26
FIGURA 10. COMPARACIÓN ENTRE SINTAXIS JAVASCRIPT Y COFFEESCRIPT.....	29
FIGURA 11. CONEXIÓN CON LA DB. ....	31
FIGURA 12. ANÁLISIS DE ENCUESTA INTERFAZ .....	102
FIGURA 13 ANÁLISIS DE PROBLEMAS.....	104
FIGURA 14. ANÁLISIS DEL GRADO DE DIFICULTAD DEL SISTEMA. ....	106
FIGURA 15. ESTRUCTURA DE ARCHIVOS PARA LA CARGA DE BOOTSTRAP .....	109
FIGURA 16. BOOTSTRAP EN NOTEBOOK.....	109
FIGURA 17. BOOTSTRAP EN UNA IPAD .....	110
FIGURA 18. BOOTSTRAP EN UN GALAXY S3.....	110

## ÍNDICE DE TABLAS

TABLA I : ACTORES DEL SISTEMA .....	43
TABLA II : PERMISOS DE USUARIOS. ....	45
TABLA III : PERMISOS DE ADMINISTRADOR.....	45
TABLA IV : PERMISOS DE ADMINISTRADOR DE FARMACIA.....	46
TABLA V: PERMISOS DE VENDEDOR. ....	46
TABLA VI: REQUERIMIENTOS NO FUNCIONALES. ....	47
TABLA VII : MIEMBROS DEL PROYECTO.....	47
TABLA VIII: HISTORIA DE USUARIO: ADMINISTRACIÓN USUARIOS.....	48
TABLA IX : HISTORIA DE USUARIO ADMINISTRACIÓN CLIENTES .....	49
TABLA X : HISTORIA DE USUARIO: ADMINISTRAR PRODUCTO .....	50
TABLA XI: HISTORIA DE USUARIO: ADMINISTAR PROVEEDORES.....	51
TABLA XII : HISTORIA DE USUARIO: ADMINISTRAR FACTURA DE VENTA .....	52
TABLA XIII : HISTORIA DE USUARIO ADMINISTRAR FACTURA DE COMPRA.....	53
TABLA XIV : HISTORIA DE USUARIO: ADMINISTRAR COMPROBANTES DE TRANSFERENCIAS DE PRODUCTO .....	54
TABLA XV : HISTORIA DE USUARIO: ADMINISTRAR RECETAS .....	55
TABLA XVI : HISTORIA DE USUARIO: CIERRE DE CAJA .....	56
TABLA XVII : HISTORIA DE USUARIO: ADMINISTRACIÓN DE INVENTARIO .....	57
TABLA XVIII : HISTORIA DE USUARIO: DESPACHO DE PRODUCTO.....	58
TABLA XIX : HISTORIA DE USUARIO: GENERAR REPORTES .....	59
TABLA XX : HISTORIA DE USUARIO: GENERAR KARDEX DE PRODUCTO.....	60
TABLA XXI : HISTORIA DE USUARIO: REPORTES DE FACTURAS DEL DÍA.....	61
TABLA XXII : HISTORIA DE USUARIO: REPORTE DE INVENTARIO MENSUAL .....	62
TABLA XXIII : HISTORIA DE USUARIO: REPORTES DE LIQUIDACIÓN MENSUAL .....	63
TABLA XXIV : RESUMEN DE HISTORIAS DE USUARIO .....	64
TABLA XXV: PLAN DE ENTREGAS. ....	65
TABLA XXVI : VELOCIDAD DEL PROYECTO .....	67
TABLA XXVII : ITERACIONES .....	67
TABLA XXVIII : GLOSARIO DE TERMINOS .....	68
TABLA XXIX : TARJETAS CRC CLIENTE .....	69
TABLA XXX : TARJETAS CRC PRODUCTO.....	69
TABLA XXXI : TARJETAS CRC PROVEEDOR .....	69
TABLA XXXII : TARJETAS CRC FACTURA DE COMPRA.....	70

TABLA XXXIII : TARJETAS CRC FACTURA DE VENTA .....	70
TABLA XXXIV : TARJETAS CRC TRANSFERENCIAS.....	70
TABLA XXXV : TARJETAS CRC HOSPITALIZACIÓN .....	70
TABLA XXXVI : TARJETAS CRC FACTURAR RECETAS .....	71
TABLA XXXVII : TARJETAS CRC CANJE.....	71
TABLA XXXVIII : TARJETAS CRC CIERRE DE CAJA .....	71
TABLA XXXIX : PRUEBA UNITARIA 1 .....	75
TABLA XL : PRUEBA UNITARIA 2.....	76
TABLA XLI : PRUEBA UNITARIA 3.....	76
TABLA XLII : PRUEBA UNITARIA 4.....	76
TABLA XLIII : CASO DE PRUEBA 1: ADMINISTRACIÓN USUARIOS.....	78
TABLA XLIV : CASO DE PRUEBA 2: ADMINISTRACIÓN USUARIOS .....	78
TABLA XLV : CASO DE PRUEBA 3: ADMINISTRACIÓN USUARIOS .....	79
TABLA XLVI : CASO DE PRUEBA 4: ADMINISTRACIÓN USUARIOS .....	79
TABLA XLVII : CASO DE PRUEBA 5: ADMINISTRACIÓN USUARIOS.....	80
TABLA XLVIII : CASO DE PRUEBA 6: ADMINISTRACIÓN CLIENTE .....	80
TABLA XLIX : CASO DE PRUEBA 7: ADMINISTRACIÓN CLIENTE.....	81
TABLA L : CASO DE PRUEBA 8: ADMINISTRACIÓN CLIENTES .....	81
TABLA LI : CASO DE PRUEBA 9: ADMINISTARCIÓN DE PROVEEDORES .....	82
TABLA LII : CASO DE PRUEBA 10: ADMINISTRACIÓN DE PROVEEDORES .....	82
TABLA LIII : CASO DE PRUEBA 11: ADMINISTRAR PROVEEDORES.....	83
TABLA LIV : CASO DE PRUEBA 12: ADMINISTRAR PRODUCTO .....	83
TABLA LV : CASO DE PRUEBA 13: ADMINISTRAR PRODUCTO .....	84
TABLA LVI : CASO DE PRUEBA 14: ADMINISTRAR FACTURA DE VENTA.....	84
TABLA LVII : CASO DE PRUEBA 15: ADMINISTRAR FACTURA DE VENTA.....	85
TABLA LVIII : CASO DE PRUEBA 16: ADMINISTRACIÓN DE FACTURAS DE VENTA .	85
TABLA LIX : CASO DE PRUEBA 17: ADMINISTRACIÓN DE TRANSFERENCIAS .....	86
TABLA LX : CASO DE PRUEBA 18: ADMINISTRACIÓN DE TRANSFERENCIAS .....	86
TABLA LXI : CASO DE PRUEBA 19: ADMINISTRACIÓN DE TRANSFERENCIAS .....	87
TABLA LXII : CASO DE PRUEBA 20: ADMINISTRACIÓN DE TRANSFERENCIAS .....	87
TABLA LXIII : CASO DE PRUEBA 21: ADMINISTRACIÓN HOSPITALIZACIÓN .....	88
TABLA LXIV : CASO DE PRUEBA 22 ADMINISTRACIÓN HOSPITALIZACIÓN .....	88
TABLA LXV : CASO DE PRUEBA 23 ADMINISTRACIÓN DE HOSPITALIZACIÓN .....	89
TABLA LXVI : CASO DE PRUEBA 24: FACTURAR RECETAS .....	89
TABLA LXVII : CASO DE PRUEBA 25: ADMINISTRAR CANJE .....	90

TABLA LXVIII : CASO DE PRUEBA 26: ADMINISTRAR CANJE .....	90
TABLA LXIX : CASO DE PRUEBA 27: CIERRE DE CAJA.....	91
TABLA LXX : CASO DE PRUEBA 28: REPORTES FINANCIEROS MENSUALES .....	91
TABLA LXXI : CASO DE PRUEBA 29: REPORTES FINANCIEROS MENSUALES .....	92
TABLA LXXII : CASO DE PRUEBA 30: REPORTES FINANCIEROS MENSUALES .....	92
TABLA LXXIII : MÓDULOS A PRUEBA.....	94
TABLA LXXIV : TALENTO HUMANO .....	112
TABLA LXXV : MATERIALES .....	112
TABLA LXXVI : HARDWARE Y SOFTWARE .....	113
TABLA LXXVII : SERVICIOS .....	114
TABLA LXXVIII : RESUMEN .....	114



## GLOSARIO

Siglas	Significado
<b>PFC</b>	Proyecto de Fin de Carrera
<b>RoR</b>	Ruby on Rails
<b>MVC</b>	Modelo, vista, controlador
<b>SOAP</b>	Simple Object Access Protocol
<b>HTML</b>	Lenguaje de marcas de hipertexto en inglés HyperText Markup Language
<b>CSS</b>	Hojas de estilo en cascada o en inglés Cascading Style Sheets
<b>XP</b>	Extreme Programming
<b>XML</b>	Lenguaje de marcas extensible, en inglés Extensible Markup Language
<b>CRC</b>	Tarjetas (Clase, Responsabilidad, Colaborador)
<b>XHTML</b>	HTML expresado como XML, en inglés extensible HyperText Markup Language
<b>JS</b>	JavaScript
<b>DOM</b>	Document, Object, Model
<b>LESS</b>	Visualizador de archivos de texto
<b>ERS</b>	Especificación de Requerimientos de Software
<b>UI</b>	Diseño de interfaz
<b>UX</b>	Experiencia de Usuario

## **c. Introducción**

Las instituciones, empresas o servicios en general son base primordial para el desarrollo y crecimiento de un país o sociedad, las mismas se destacan en diferentes campos ya sea a nivel público o privado, siempre brindan servicios al mundo en general, muchas de estas sociedades ven el gran valor que tiene la tecnología que día a día avanza a pasos agigantados. Dentro de instituciones o empresas tenemos al **HOSPITAL BÁSICO 7 BI LOJA**, regentado por el estado y con cobertura a nivel nacional, brindando en este caso servicio de calidad en la ciudad de Loja.

Es así, que la creciente necesidad de dar mejor eficiencia y rapidez a los clientes y personal que labora en la institución; en cuanto a lo que se realiza en farmacia obligan a esta entidad, a desarrollar un sistema que eleve el nivel de calidad de los servicios, esto con la finalidad de llevar un inventario ordenado, reportes diarios de compra y ventas, proveedores y productos.

Basado en los avances técnico-tecnológicos, como son los sistemas web de gestión de farmacia, que por medios diferentes ofrecen iguales servicios a otras empresas o instituciones se tiene como objetivo el: “Sistema web de gestión de farmacia basado en tecnología móvil para el hospital básico 7 B.I Loja”, con el propósito de que el hospital cuente con un sistema eficiente que ayude al mejor manejo y control de información y productos dentro de farmacia.

Como solución a los problemas antes mencionados se ha desarrollado módulos que ayuden al buen uso de la información, como el correcto desempeño de las tareas que tienen los usuarios de farmacia y así lograr el correcto funcionamiento del sistema en los departamentos de estadística y central de enfermería.

La investigación desarrollada se encuentra bajo los parámetros establecidos por la Universidad Nacional de Loja y el Área de la Energía las Industrias y los Recursos Naturales No Renovables de la siguiente manera:

Resumen que describe una síntesis general del contenido de la tesis, el que adicional se encuentra en inglés cuya traducción ha sido validada por un profesional en la materia (ver anexo 9), continua con introducción que explica la importancia real que traerá consigo este sistema. En la metodología se establece los métodos de investigación tanto científicos, experimentales y técnicas de investigación, así también las fases que se utilizaron para el desarrollo de la tesis.

En la revisión de literatura se presenta información de los diferentes aspectos que se tomaron en cuenta para llevar a cabo esta tesis en 3 capítulos.

En el primer capítulo se detalla información relacionada con el framework Ruby on Rails mismo que es esencial para el desarrollo del sistemas web de gestión, en el segundo capítulo se estudió de la metodología XP, con el objetivo de elaborar el correcto funcionamiento del sistema; y el capítulo 3 se basa en herramientas de desarrollo para la codificación del sistema.

De esta forma se da a conocer la interrogante del problema: ¿Al aplicar la herramienta Ruby on Rails se mejorará el servicio de farmacia para el Hospital Básico 7 B.I "Loja"?

## **d. Revisión de Literatura.**

### **Capítulo 1**

#### **1. Ruby on Rails**

##### **1.1.¿Qué es Ruby?**

Ruby es un Lenguaje de Programación de alto nivel orientado a objetos en el cual hasta los números y los caracteres literales son objetos y tienen métodos de su clase, que pueden llamarse normalmente, este es rápido y sencillo, por lo cual su línea de aprendizaje no es tan extensa a diferencia de otros lenguajes como: java, python, php, etc.

##### **1.2.Introducción.**

Ruby, es un lenguaje creado por Yukihiro "Matz" Matsumoto, quien empezó a trabajar en Ruby el 24 de febrero de 1993, y lo presentó al público en el año 1995.

El cual ha manifestado que está “tratando de hacer que Ruby sea natural, no simple”, de una forma que se asemeje a la vida real, Ruby es simple en apariencia, pero complejo por dentro, como el cuerpo humano.

Además que está diseñado para la productividad y la diversión del desarrollador, siguiendo los principios de una buena interfaz de usuario, el diseño de sistemas necesita enfatizar las necesidades humanas más que las de la máquina.

En Ruby, todo es un objeto. Se le puede asignar propiedades y acciones a toda información y código. La programación orientada a objetos llama a las propiedades variables de instancia y las acciones son conocidas como métodos. La orientación a objetos pura de Ruby se suele demostrar con un simple código que aplica una acción a un número. En muchos lenguajes, los números y otros tipos

primitivos no son objetos. Ruby sigue la influencia del lenguaje Smalltalk pudiendo poner métodos y variables de instancia a todos sus tipos de datos. Esto facilita el uso de Ruby, porque las reglas que se aplican a los objetos son aplicables a todo Ruby [1].

La última versión estable de la rama 1.8 es la 1.8.7\_p248, de la rama 1.9 es la 1.9.1\_p378. La versión en 1.9 que incorpora mejoras sustanciales en el rendimiento del lenguaje, que se espera queden reflejadas en la próxima versión estable de producción del lenguaje, Ruby 1.9.0.1 Diferencias en rendimiento entre la actual implementación de Ruby (1.8.6) y otros lenguajes de programación más arraigados han llevado al desarrollo de varias máquinas virtuales para Ruby. Entre éstas se encuentra JRuby, un intento de llevar Ruby a la plataforma Java, y Rubinius, un intérprete modelado basado en las máquinas virtuales de Smalltalk. Los principales desarrolladores han apoyado la máquina virtual proporcionada por el proyecto YARV, que se fusionó en el árbol de código fuente de Ruby el 31 de diciembre de 2006, y se dio a conocer como Ruby 1.9.

### **1.3. Características de Ruby**

Al ser Ruby un lenguaje orientado a objetos y bastante flexivo posee muchas características las cuales se mencionan a continuación:

Lenguaje de guiones interpretado:

1. Posibilidad de realizar directamente llamadas al sistema operativo.
2. Potentes operaciones sobre cadenas de caracteres y expresiones regulares.
3. Retroalimentación inmediata durante el proceso de desarrollo.

Programación orientada a objetos:

1. Todo es un objeto
2. Clases, herencia, métodos.
3. Mixins por módulos
4. Iteradores y cierres.

Rápido y sencillo:

1. Son innecesarias las declaraciones de variables.
2. Las variables no tienen tipo
3. La sintaxis es simple y consistente
4. La gestión de la memoria es automática.

## **1.4. Ruby on Rails como herramienta de desarrollo de aplicaciones web.**

### **1.4.1 Introducción.**

Ruby on Rails, también conocido como RoR o Rails es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros frameworks y con un mínimo de configuración. El lenguaje de programación Ruby permite la metaprogramación, de la cual Rails hace uso, lo que resulta en una sintaxis que muchos de sus usuarios encuentran muy legible. Rails se distribuye a través de RubyGems, que es el formato oficial de paquete y canal de distribución de bibliotecas y aplicaciones Ruby [2].

Para llevar a cabo una instalación cien por ciento segura tomamos en cuenta la página oficial de Rails [3].

### 1.4.2 Arquitectura de Ruby on Rails

La arquitectura que maneja es la de MVC (Modelo, vista, controlador) el cual lo podemos observar en la siguiente Figura1:

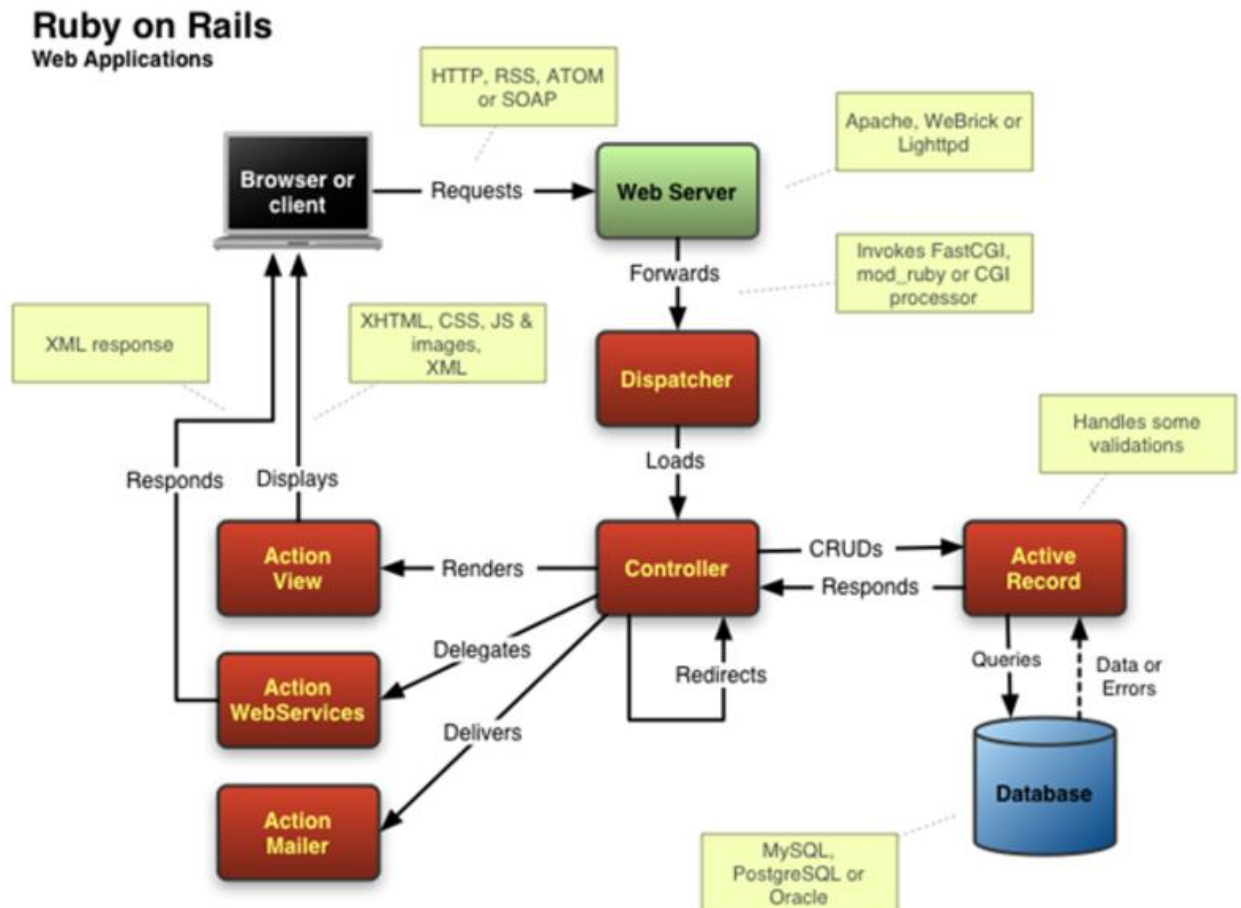


Figura 1. Arquitectura de una aplicación desarrollada con RoR [4]

## **Modelo.**

Las aplicaciones web orientadas a objetos sobre bases de datos, el Modelo consiste en las clases que representan a las tablas de la base de datos.

En Ruby on Rails, las clases del Modelo son gestionadas por ActiveRecord. Por lo general, lo único que tiene que hacer el programador es heredar de la clase ActiveRecord: Base, y el programa averiguará automáticamente qué tabla usar y qué columnas tiene.

Las definiciones de las clases también detallan las relaciones entre clases con sentencias de mapeo objeto relacional. Por ejemplo, si la clase "Persona tiene una definición has\_many: facturas, y existe una instancia de persona llamada a entonces p.facturas devolverá un array con todos los objetos Factura cuya columna persona\_id, sea igual a p.id.

Las rutinas de validación de datos (p.e. validates\_uniqueness\_of: checksum) y las rutinas relacionadas con la actualización (p.e. after\_destroy:borrar\_archivo, before\_update:actualizar\_detalle) también se especifican e implementan en la clase del modelo.

Lo que realiza el modelo al final no es más que las tablas de la base de datos y sus relaciones, además de las migraciones que son los cambios que se realizan en la BD.

## **Vista.**

La Vista es la lógica de visualización, o cómo se muestran los datos de las clases del Controlador. Con frecuencia en las aplicaciones web la vista consiste en una cantidad mínima de código incluido en HTML.

El método que se emplea en Rails por defecto es usar Ruby Embebido (archivos.rhtml, desde la versión 2.x en adelante de RoR archivos.html.erb), que son básicamente fragmentos de código HTML con algo de código en Ruby, siguiendo una sintaxis similar a JSP. También pueden construirse vistas en HTML y XML con Builder o usando el sistema de plantillas Liquid.

Es necesario escribir un pequeño fragmento de código en HTML para cada método del controlador que necesita mostrar información al usuario. El "maquetado" o distribución



de los elementos de la página se describe separadamente de la acción del controlador y los fragmentos pueden invocarse unos a otros.

### **Controlador.**

En MVC, las clases del Controlador responden a la interacción del usuario e invocan a la lógica de la aplicación, que a su vez manipula los datos de las clases del Modelo y muestra los resultados usando las Vistas. En las aplicaciones web basadas en MVC, los métodos del controlador son invocados por el usuario usando el navegador web.

La implementación del Controlador es manejada por el ActionPack de Rails, que contiene la clase ApplicationController. Una aplicación Rails simplemente hereda de esta clase y define las acciones necesarias como métodos, que pueden ser invocados desde la web, por lo general en la forma:

`http://aplicacion/ejemplo/metodo`

Esta invoca a `EjemploController#método`, y presenta los datos usando el archivo de plantilla `/App/views/ejemplo/método.html.erb`, a no ser que el método redirija a algún otro lugar.

Rails también proporciona andamiaje, que puede construir rápidamente la mayor parte de la lógica y vistas necesarias para realizar las operaciones más frecuentes.

La figura 2 indica el Modelo Vista Controlador en Ruby on Rails.

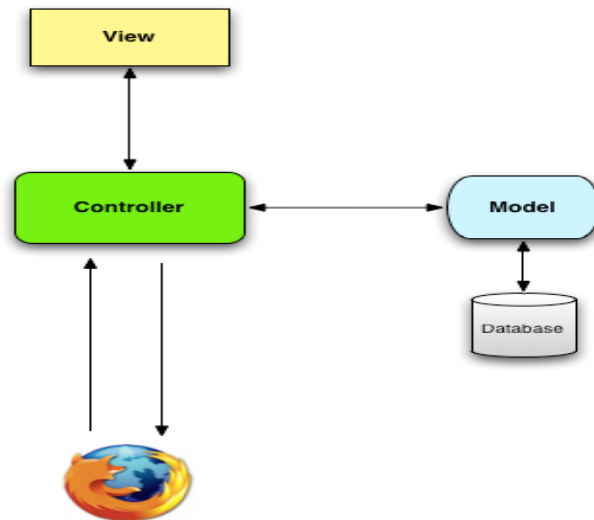


Figura 2. Modelo, vista, controlador en RoR

## Otros módulos

Además, Rails ofrece otros módulos, como Action Mailer (para enviar correo electrónico) o Active Resource que proporciona la infraestructura necesaria para crear de manera sencilla recursos REST, algo por lo que apuesta claramente Rails en sus últimas versiones desplazando así a otros modelos como SOAP y XML-RPC a los que se les daba soporte en versiones anteriores mediante Action Web Service.

## 1.5. Desarrollo de Aplicaciones

### 1.5.1. Descripción de Herramientas afines

Hay muchas alternativas para trabajar con Ruby on Rails, tanto libres y gratuitas como de pago. A continuación se listan las principales:

1. Aptana: Multiplataforma. Nació como plugins de eclipse para la edición y desarrollo web. Actualmente puedes instalarlo como plugins o autónomo de

forma independiente. Las últimas versiones están muy bien integradas con Ruby on Rails. En este momento Aptana 3 está en una versión BETA.

2. Netbeans: Uno de los más usados, libre y totalmente gratuito. Viene muy bien integrado con JRuby (lo cual es algo lógico pues es un programa de Sun).
3. TextMate: Sólo para Mac. Es el entorno más usado entre la comunidad Rails. Es pago pero su potencia y forma de trabajo favorece la producción y desarrollo con Ruby on Rails.
4. Sublime Text: Multiplataforma, es un editor de texto de código fuente, creado por phytom el cual se distribuye de forma libre pero no es software libre. Actualmente es uno de los más utilizados por los desarrolladores.

Cabe indicar que los casos de éxito de Ruby son:

1. Twitter.- La red social más famosa y consciente en el mundo del microblogging, creado por Jack Dorsey en el 2006, en california EE UU.
1. Github.- El repositorio que ayuda alojar proyectos de forma pública y privada es de las más utilizadas por los desarrollares del mundo, creado por la empresa hitgub Inc., lanzado en el año 2008.
2. Shopify.- Es una solución de sitio web de comercio electrónico que le permite vender en línea al darle todo lo que necesita para crear su tienda online, creado por la empresa shopify. Inc en el año 2006.
3. Hospitium.- Software para la gestión de refugios de animales, rescates, creado por Travis Berry y Jinna Adansom en el año de 2010.
4. How are you?- Es una red social por la salud que pone a los pacientes en contacto con su médico en tiempo real, creada por la empresa Cambridge healthcare en el año 2013.

## **COMENTARIO.**

Es una herramienta fácil de usar que complementa el tipo de desarrollo que se realizó en este caso el sistema web de farmacia, es una forma de desarrollar proyectos en otro tipo de ámbito ya se sale de lo normal que es java, Python, php entre otros.

Lo importante de esta herramienta es que genera todo lo necesario para empezar a trabajar en proyectos grandes.

## **Capítulo 2.**

### **2. Metodología de desarrollo XP.**

#### **2.1. Programación Extrema (Extreme Programming, XP)**

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.

XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. Kent Beck, el padre de XP, describe la filosofía de XP en [5] sin cubrir los detalles técnicos y de implantación de las prácticas. Posteriormente, otras publicaciones de experiencias se han encargado de dicha tarea. A continuación presentaremos las características esenciales de XP organizadas en los tres apartados siguientes: historias de usuario, roles, proceso y prácticas.

## **2.2. Roles XP.**

Aunque en otras fuentes de información aparecen algunas variaciones y extensiones de roles XP, en este apartado describiremos los roles de acuerdo con la propuesta original de Beck.

### **2.2.1. Programador**

El programador escribe las pruebas unitarias y produce el código del sistema. Debe existir una comunicación y coordinación adecuada entre los programadores y otros miembros del equipo.

### **2.2.2. Cliente**

El cliente escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio. El cliente es sólo uno dentro del proyecto pero puede corresponder a un interlocutor que está representando a varias personas que se verán afectadas por el sistema.

### **2.2.3. Encargado de pruebas (Tester)**

El encargado de pruebas ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.

### **2.2.4. Encargado de seguimiento (Tracker)**

El encargado de seguimiento proporciona realimentación al equipo en el proceso XP. Su responsabilidad es verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, comunicando los resultados para mejorar futuras estimaciones.

También realiza el seguimiento del progreso de cada iteración y evalúa si los objetivos son alcanzables con las restricciones de tiempo y recursos presentes. Determina cuándo es necesario realizar algún cambio para lograr los objetivos de cada iteración.

### **2.2.5. Entrenador (Coach)**

Es responsable del proceso global. Es necesario que conozca a fondo el proceso XP para proveer guías a los miembros del equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.

### **2.2.6. Consultor**

Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto. Guía al equipo para resolver un problema específico.

### **2.2.7. Gestor (Big boss)**

Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

## **2.3. Principios Básicos**

La Programación Extrema se basa en 12 principios básicos agrupados en cuatro categorías, a continuación se presenta cada una de ellas, detallada de cómo se llevó a cabo su cumplimiento en el desarrollo de la tesis:

### **2.3.1. Retroalimentación a escala fina.**

**1. El principio de pruebas:** Se tiene que establecer un período de pruebas de aceptación del programa. Dentro del desarrollo del sistema se utilizó la herramienta Unit\_Test del framework CodeIgniter, en la fase 3 de resultados se encuentra el detalle.

**2. Proceso de planificación:** Es un espacio frecuente de comunicación entre el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración. Esta práctica se puede ilustrar como un juego, donde existen dos tipos de jugadores: Cliente y Programador. El cliente establece la prioridad de cada historia de usuario, de acuerdo con el valor que aporta para el negocio. Los programadores estiman el esfuerzo asociado a cada

historia de usuario. Se ordenan las historias de usuario según prioridad y esfuerzo, y se define el contenido de la entrega y/o iteración, apostando por enfrentar lo de más valor y riesgo cuanto antes. Este juego se realiza durante la planificación de la entrega, en la planificación de cada iteración y cuando sea necesario reconducir el proyecto.

**3. El cliente en el sitio:** Durante el desarrollo del sistema se mantuvo la comunicación con el usuario permitiendo determinar los requerimientos, definir la funcionalidad, señalar las prioridades.

El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Gran parte del éxito del proyecto XP se debe a que es el cliente quien conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita, ya que esta última toma mucho tiempo en generarse y puede tener más riesgo de ser mal interpretada. En [6] Jeffries indica que se debe pagar un precio por perder la oportunidad de un cliente con alta disponibilidad. Algunas recomendaciones propuestas para dicha situación son las siguientes: intentar conseguir un representante que pueda estar siempre disponible y que actúe como interlocutor del cliente, contar con el cliente al menos en las reuniones de planificación, establecer visitas frecuentes de los programadores al cliente para validar el sistema, anticiparse a los problemas asociados estableciendo llamadas telefónicas frecuentes y conferencias, reforzando el compromiso de trabajo en equipo.

**4. Programación en parejas:** Toda la producción de código debe realizarse con trabajo en parejas de programadores.

Según Cockburn y Williams en un estudio realizado para identificar los costos y beneficios de la programación en parejas [5], las principales ventajas de introducir este estilo de programación son: muchos errores son detectados conforme son introducidos en el código (inspecciones de código continuas), por consiguiente la tasa de errores del producto final es más baja, los diseños son mejores y el tamaño del código menor (continúa discusión de ideas de los programadores), los problemas de programación se resuelven más rápido, se posibilita la transferencia de conocimientos de programación entre los miembros del equipo, varias personas entienden las diferentes partes sistema, los programadores conversan mejorando así el flujo de información y la dinámica del equipo, y finalmente, los programadores disfrutan más su trabajo.

Dichos beneficios se consiguen después de varios meses de practicar la programación en parejas. En los estudios realizados por Cockburn y Williams este lapso de tiempo varía de 3 a 4 meses.

### **2.3.2. Proceso continuo en lugar de por lotes.**

**1. Integración continua:** Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día. Todas las pruebas son ejecutadas y tienen que ser aprobadas para que el nuevo código sea incorporado definitivamente. La integración continua a menudo reduce la fragmentación de los esfuerzos de los desarrolladores por falta de comunicación sobre lo que puede ser reutilizado o compartido. Martin Fowler en [6] afirma que el desarrollo de un proceso disciplinado y automatizado es esencial para un proyecto controlado, el equipo de desarrollo está más preparado para modificar el código cuando sea necesario, debido a la confianza en la identificación y corrección de los errores de integración.

**2. Refactorización:** La refactorización es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. La refactorización mejora la estructura interna del código sin alterar su comportamiento externo [6]. Robert Martin señala que el diseño del sistema de software es una cosa viviente. No se puede imponer todo en un inicio, pero en el transcurso del tiempo este diseño evoluciona conforme cambia la funcionalidad del sistema. Para mantener un diseño apropiado, es necesario realizar actividades de cuidado continuo durante el ciclo de vida del proyecto.

De hecho, este cuidado continuo sobre el diseño es incluso más importante que el diseño inicial. Un concepto pobre al inicio puede ser corregido con esta actividad continua, pero sin ella, un buen diseño inicial se degradará.

**3. Entregas pequeñas:** La idea es producir rápidamente versiones del sistema que sean operativas, aunque obviamente no cuenten con toda la funcionalidad pretendida para el sistema pero sí que constituyan un resultado de valor para el negocio. Una entrega no debería tardar más 3 meses.



### 2.3.3. Entendimiento compartido.

**1. Diseño simple:** Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto. La complejidad innecesaria y el código extra debe ser removido inmediatamente. Kent Beck dice que en cualquier momento el diseño adecuado para el software es aquel que: supera con éxito todas las pruebas, no tiene lógica duplicada, refleja claramente la intención de implementación de los programadores y tiene el menor número posible de clases y métodos.

**2. Metáfora:** El sistema fue desarrollado por módulos, cada módulo comprende algunas funciones, de tal forma que se permita acoplar a cada tipo de usuario dentro del sistema. Estas funciones aparecieron al ir elaborando las historias de usuario, tomando en cuenta las tarjetas CRC, las mismas que ayudarán a definir las actividades durante el diseño del sistema.

En XP no se enfatiza la definición temprana de una arquitectura estable para el sistema. Dicha arquitectura se asume evolutiva y los posibles inconvenientes que se generarían por no contar con ella explícitamente en el comienzo del proyecto se solventan con la existencia de una metáfora. El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema. Martin Fowler en [4] explica que la práctica de la metáfora consiste en formar un conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema. Este conjunto de nombres ayuda a la nomenclatura de clases y métodos del sistema.

**3. Propiedad colectiva del código:** un código con propiedad compartida. Nadie es el propietario de nada, todos son el propietario de todo [7].

Cualquier programador puede cambiar cualquier parte del código en cualquier momento. Esta práctica motiva a todos a contribuir con nuevas ideas en todos los segmentos del sistema, evitando a la vez que algún programador sea imprescindible para realizar cambios en alguna porción de código.

**4. Estándar de codificación:** define la propiedad del código compartido así como las reglas para escribir y documentar el código y la comunicación entre diferentes piezas de código desarrolladas por diferentes equipos. Respecto a este punto no se encontró ningún problema, además fue desarrollado por una sola persona.

XP enfatiza la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación (del equipo, de la organización u otros estándares reconocidos para los lenguajes de programación utilizados). Los estándares de programación mantienen el código legible para los miembros del equipo, facilitando los cambios.

#### **2.3.4. Bienestar del programador.**

**1. La semana de 40 horas:** Se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo. Los proyectos que requieren trabajo extra para intentar cumplir con los plazos suelen al final ser entregados con retraso. En lugar de esto se puede realizar el juego de la planificación para cambiar el ámbito del proyecto o la fecha de entrega.

#### **2.4. Fases de la Metodología**

La metodología XP consta de cuatro fases tal como lo indica la figura 3, para el desarrollo de la aplicación se utilizó cada una de ellas.

La relación entre el equipo de diseño, los que desarrollan el software y clientes es totalmente diferente al que se produce en las metodologías tradicionales, que se basaba en una fase de captura de los requisitos previa al desarrollo, y de una fase de validación posterior al mismo [8].

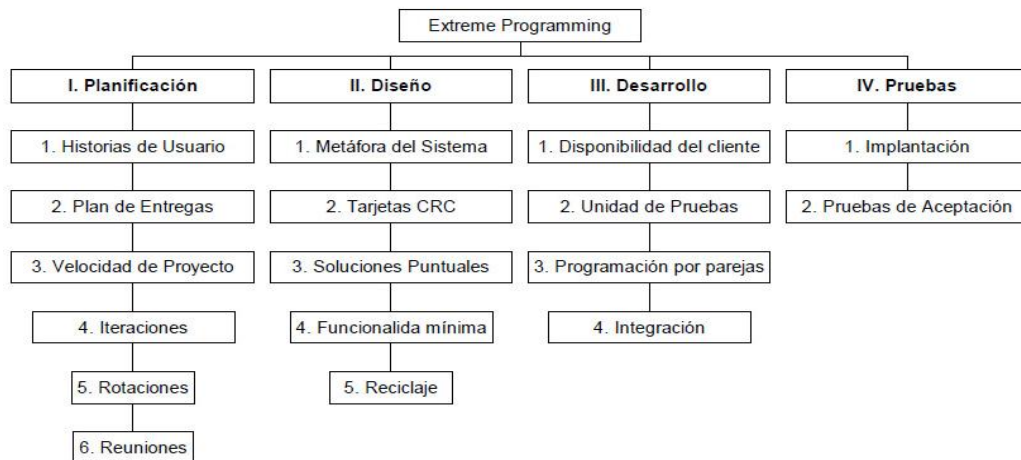


Figura 3. Fases de la Metodología XP (aplicada) [8]

### 2.4.1. FASE 1: Planificación del Proyecto

#### 1. Historias de Usuario:

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales.

El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento historias de usuario pueden romperse, remplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas [7].

Respecto de la información contenida en la historia de usuario, existen varias plantillas sugeridas pero no existe un consenso al respecto. En muchos casos sólo se propone utilizar un nombre y una descripción o sólo una descripción, más quizás una estimación de esfuerzo en días. Beck en su libro presenta un ejemplo de ficha (customer story and task card) en la cual pueden reconocerse los siguientes contenidos:

Fecha, tipo de actividad (nueva, corrección, mejora), prueba funcional, número de historia, prioridad técnica y del cliente, referencia a otra historia previa, riesgo, estimación técnica, descripción, notas y una lista de seguimiento con la fecha, estado cosas por terminar y comentarios.

Una de las interrogantes (que también se presenta cuando se utilizan casos de uso) es ¿cuál es el nivel de granularidad adecuado para una historia de usuario? La respuesta no es tajante. Jeffries en [6] dice que depende de la complejidad del sistema, debe haber al menos una historia por cada característica importante, y propone realizar una o dos historias por programador por mes. Si se tienen menos, probablemente sea conveniente dividir las historias, si se tienen más lo mejor es disminuir el detalle y agruparlas. Para efectos de planificación, las historias pueden ser de una a tres semanas de tiempo de programación (para no superar el tamaño de una iteración).

No hay que preocuparse si en un principio no se identifican todas las historias de usuario. Al comienzo de cada iteración estarán registrados los cambios en las historias de usuario y según eso se planificará la siguiente iteración.

Las historias de usuario son descompuestas en tareas de programación y asignadas a los programadores para ser implementadas durante una iteración.

## **2. Planes de Entrega:**

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días.

Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto, equivale a una semana ideal de programación. Las historias generalmente valen de 1 a 3 puntos. Por otra parte, El equipo de desarrollo mantiene un registro de la “velocidad” de desarrollo, establecida en puntos por iteración, basándose principalmente en la suma de puntos correspondientes a las historias de usuario que fueron terminadas en la última iteración.

La planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación.

### **3. Iteraciones:**

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide qué historias se implementarán en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción.

Los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores.

#### **4. La Velocidad del Proyecto:**

Medida que representa la rapidez con la que se desarrolla el proyecto; estimarla es muy sencillo, basta con contar el número de historias de usuario que se pueden implementar en una iteración; de esta forma, se sabrá el cupo de historias que se pueden desarrollar en las distintas iteraciones. Usando la velocidad del proyecto controlaremos que todas las tareas se puedan desarrollar en el tiempo del que dispone la iteración. Es conveniente reevaluar esta medida cada 3 ó 4 iteraciones y si se aprecia que no es adecuada hay que negociar con el cliente un nuevo "Release Plane" [8].

#### **2.4.2. FASE 2: Diseño.**

##### **1. Diseños Simples**

Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto. La complejidad innecesaria y el código extra debe ser removido inmediatamente. Kent Beck dice que en cualquier momento el diseño adecuado para el software es aquel que: supera con éxito todas las pruebas, no tiene lógica duplicada, refleja claramente la intención de implementación de los programadores y tiene el menor número posible de clases y métodos.

##### **2. Glosarios de Términos**

Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código.

##### **3. Refactorizar**

Refactorizar es mejorar y modificar la estructura y codificación de códigos ya creados sin alterar su funcionalidad. Refactorizar supone revisar de nuevo estos códigos para procurar optimizar su funcionamiento. Es muy común rehusar códigos ya creados que contienen funcionalidades que no serán usadas y diseños obsoletos.

### **2.4.3. FASE 3: Desarrollo.**

Antes del desarrollo de cada historia de usuario el cliente debe especificar detalladamente lo que ésta hará y también tendrá que estar presente cuando se realicen los test que verifiquen que la historia implementada cumple la funcionalidad especificada. La codificación debe hacerse atendiendo a estándares de codificación ya creados.

### **2.4.4. FASE 4: Pruebas.**

La producción de código está dirigida por las pruebas unitarias. Las pruebas unitarias son establecidas antes de escribir el código y son ejecutadas constantemente ante cada modificación del sistema. Los clientes escriben las pruebas funcionales para cada historia de usuario que deba validarse. En este contexto de desarrollo evolutivo y de énfasis en pruebas constantes, la automatización para apoyar esta actividad es crucial.

## **Capítulo 3.**

### **3. Herramientas de desarrollo.**

#### **3.1. JAVASCRIPT.**

JavaScript (abreviado comúnmente "JS") es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo [9].

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Un ejemplo de código en javascript se lo puede observar en la figura 4, que describe como cargar las librerías de JQuery, Bootstrap entre otras.

```
1 // This is a manifest file that'll be compiled into application.js, which will include all the files
2 // listed below.
3 //
4 // Any JavaScript/Coffee file within this directory, lib/assets/javascripts, vendor/assets/javascripts,
5 // or vendor/assets/javascripts of plugins, if any, can be referenced here using a relative path.
6 //
7 // It's not advisable to add code directly here, but if you do, it'll appear at the bottom of the
8 // compiled file.
9 //
10 // Read Sprockets README (https://github.com/sstephenson/sprockets#sprockets-directives) for details
11 // about supported directives.
12 //
13 //= require jquery
14 //= require jquery_ujs
15 //= require turbolinks
16 //= require bootstrap
17 //= require_tree .
18 //= require jquery_nested_form
19 //= require nested
20 //= require nested2
21 //= require alertify
22 //= require chosen.jquery
23 //= require moment
24 //= require jquery.dataTables
25 //= require bootstrap-datepicker
26 //= require jquery-ui
27 //= require best_in_place
28 //= require timepicker
29
```

Figura 4. Código para cargar las librerías en JavaScript

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML.

### 3.2. CSS (Hojas de estilo en cascada)

El lenguaje HTML está limitado a la hora de aplicarle forma a un documento. Esto es así porque fue concebido para otros usos, distintos a los actuales, mucho más amplios. Para mejorar la apariencia de las páginas se usaron distintas vías: Extensiones propietarias del HTML (soluciones propias de Netscape y de Microsoft), conversión de texto en imágenes de texto para hacerle mantener el mismo aspecto en cualquier navegador, imágenes para controlar los espacios en blanco y otros efectos, tablas para controlar el diseño global de la página, programas o scripts en lugar del



marcado HTML. Estas técnicas aumentan considerablemente la complejidad de las páginas web, tienen escasa flexibilidad y problemas de compatibilidad [12].

Mediante etiquetas HTML en una página Web se define la estructura del documento y con las definiciones de estilos se describe la apariencia. Las hojas de estilo en cascada, definen el efecto visual que tendrán las distintas etiquetas HTML. CSS3 es el último estándar de CSS [12].

La figura 5 muestra un ejemplo de codificación en CSS de un datatable, para cambiar el fondo a color blanco y borde de un pixel.

```
table.dataTable td.sorting_1{ background-color: white; border:1px lightgrey; }
table.dataTable td{ background-color: white; border:1px lightgrey;}
table.dataTable tr.odd { background-color: white; border:1px lightgrey;}
table.dataTable tr.even{ background-color: white; border:1px lightgrey; }
```

Figura 5. Código CSS

La figura 6 muestra un datatable normal y la figura 7 muestra un datatable aplicando estilo con el código anterior de la figura 6.

Mostrar	10	registros	Buscar:	
Nivel	▲	Evaluación	◇	Nota
Nada en esta tabla				
Mostrando registros del 0 al 0 de un total de 0 registros				

Figura 6. Datatable normal

Nombre	Identificación	Dirección	Teléfono	Email	Acciones
Amanda Díaz Orozco	1101109344	La Argelia 34-98	095643890	amanda17@hotmail.com	 
Angel Alberto Valdez	1702998558	Clodoveo	2578414		 
Angel Valdez	1103786990	Loja	098662398		 
Carla Medina	1104598451	Loja			 
Carrillo Hinogosa Orlando Mauricio	1715622732	Barrio El Pedestal	2787987		 
Daniel Francisco Reinoso Ortega	1100310216	Miguel Riofrio Y Sucre	2195167		 
Demo	1104369515				 
Enrique Eduardo Torres Torres	1101097671	San Sebastian 12-23	0923929323		 
Fabio Calvache	1103082044	Clodoveo	2574789		 
Fabricio Flores	1104015936	Loja	098765345		 

Mostrando registros del 1 al 10 de un total de 20 registros (filtrado de un total de 21 registros)

Figura 7. Datatable aplicado estilo

### **3.3. JQUERY**

JQuery es un framework Javascript libre y open source, del lado del cliente; el objetivo de esta librería de java script es simplificar los comandos comunes de javascript. En pocas palabras escribir menos para hacer más.

#### **3.3.1. Los puntos fuertes de jquery**

El framework JQuery es cada vez más aceptables por los grandes desarrollares del mundo, ya que las aportaciones de este entorno son numerosas.

El enfoque de JQuery no consiste solo en una codificación de los scripts más intuitiva y concisa, sino que su filosofía es concentrarse en el conjunto de los elementos que gestiona el DOM (Documentación, objetos y modelo) [10].

De esta manera la librería JQuery se carga cada vez que es necesario para el correcto funcionamiento de la página.

### **3.4. BOOTSTRAP**

#### **3.4.1. Introducción**

Bootstrap es un framework o conjunto de herramientas de software libre para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

#### **3.4.2. Características**

Bootstrap tiene un soporte relativamente incompleto para HTML5 y CSS 3, pero es compatible con la mayoría de los navegadores web. La información básica de compatibilidad de sitios web o aplicaciones está disponible para todos los dispositivos móviles y navegadores. Existe un concepto de compatibilidad parcial que hace disponible la información básica de un sitio web para todos los dispositivos móviles y navegadores [13].

Un ejemplo real de lo que se pudo realizar con Bootstrap es el diseño responsive que hace que se adapte al tamaño o pulgadas de cualquier dispositivo o laptop tal como lo muestra la figura 8 y Figura 9.

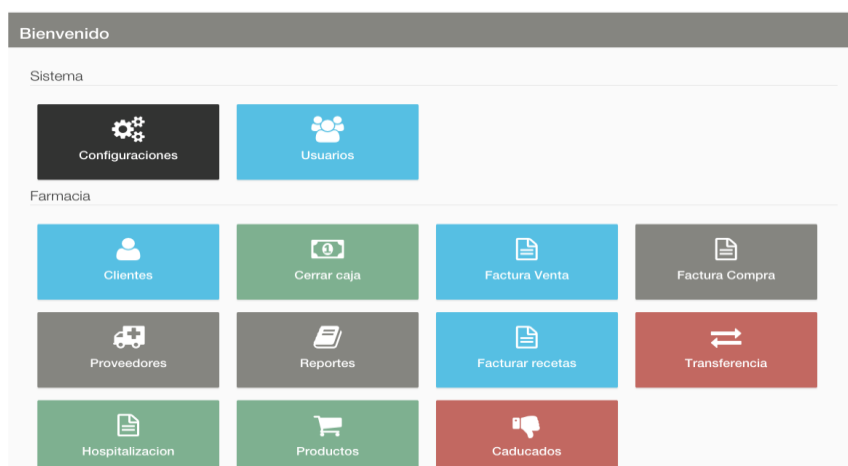


Figura 8. Tamaño completo en una laptop.



Figura 9. Tamaño desde un dispositivo móvil

Desde la versión 2.0 también soporta diseños sensibles. Esto significa que el diseño gráfico de la página se ajusta dinámicamente, tomando en cuenta las características del dispositivo usado (Computadoras, tabletas, teléfonos móviles). Bootstrap es de código abierto y está disponible en GitHub.

### **3.4.3. Estructura y función.**

Bootstrap es modular y consiste esencialmente en una serie de hojas de estilo LESS que implementan la variedad de componentes de la herramienta. Una hoja de estilo llamada `bootstrap.less` incluye los componentes de las hojas de estilo. Los desarrolladores pueden adaptar el mismo archivo de Bootstrap, seleccionando los componentes que deseen usar en su proyecto.

Los ajustes son posibles en una medida limitada a través de una hoja de estilo de configuración central. Los cambios más profundos son posibles mediante las declaraciones LESS [13].

El uso del lenguaje de hojas de estilo LESS permite el uso de variables, funciones y operadores, selectores anidados, así como clases mixin.

Desde la versión 2.0, la configuración de Bootstrap también tiene una opción especial de "Personalizar" en la documentación. Por otra parte, los desarrolladores eligen en un formulario los componentes y ajustes deseados, y de ser necesario, los valores de varias opciones a sus necesidades. El paquete consecuentemente generado ya incluye la hoja de estilo CSS pre-compilada.

#### **3.4.3.1. Sistema de cuadrilla y diseño sensible.**

Bootstrap viene con una disposición de cuadrilla estándar de 940 píxeles de ancho. Alternativamente, el desarrollador puede usar un diseño de ancho-variable. Para ambos casos, la herramienta tiene cuatro variaciones para hacer uso de distintas resoluciones y tipos de dispositivos: teléfonos móviles, formato de retrato y paisaje, tabletas y computadoras con baja y alta resolución (pantalla ancha). Esto ajusta el ancho de las columnas automáticamente.

### **3.4.3.2. Entendiendo la hoja de estilo CSS**

Bootstrap proporciona un conjunto de hojas de estilo que proveen definiciones básicas de estilo para todos los componentes de HTML. Esto otorga una uniformidad al navegador y al sistema de anchura, da una apariencia moderna para el formateo de los elementos de texto, tablas y formularios.

### **3.4.3.3. Componentes re-usables**

En adición a los elementos regulares de HTML, Bootstrap contiene otra interfaz de elementos comúnmente usados. Ésta incluye botones con características avanzadas (e.g grupo de botones o botones con opción de menú desplegable, listas de navegación, etiquetas horizontales y verticales, ruta de navegación, paginación, etc.), etiquetas, capacidades avanzadas de miniaturas tipográficas, formatos para mensajes de alerta y barras de progreso [13].

## **3.5. COFFEESCRIPT**

CoffeeScript es un lenguaje de programación que se compila a JavaScript. El lenguaje añade azúcar sintáctico inspirado en Ruby, Python y Haskell para mejorar la brevedad y la legibilidad de JavaScript, y añade características más sofisticadas, como la comprensión de listas y la coincidencia de patrones. CoffeeScript compila previsiblemente a JavaScript y los programas se pueden escribir por lo general en 1/3 menos de líneas de código sin ningún efecto en el rendimiento ni en el tiempo de ejecución. A partir del 16 de marzo de 2011, CoffeeScript ha estado en la lista de los proyectos más visitados en GitHub, y al 27 de julio 2012 es el undécimo lenguaje más popular en GitHub [11].

El lenguaje tiene un seguimiento relativamente grande por parte de la comunidad Ruby. El soporte a CoffeeScript está incluido en Ruby on Rails desde su versión 3.1.

### 3.5.1. Sintaxis

Muchas sentencias de JavaScript se pueden utilizar como expresiones en CoffeeScript, por ejemplo `if`, `switch` y `for`. Estas declaraciones de control también tienen versiones de Postfix. Un principio general es que muchos de los paréntesis son innecesarios y se pueden omitir ya que la sangría puede ser utilizada en lugar de la llaves para denotar bloques de código, las llamadas a funciones son implícitas (los paréntesis necesarios para una llamada a una función en JavaScript se puede quitar), los literales de objetos son a menudo detectados automáticamente, y así sucesivamente.

La figura 10 muestra el trabajo de la sintaxis con tabulación o sangría donde determina como es la sentencia en CoffeeScript y en JavaScript; y podemos observar que al lado derecho es CoffeeScript e indica que no es necesario poner paréntesis como lo explica el párrafo anterior.

```
1 window.Helpers || (window.Helpers = {});  
2 ~  
3 window.Helpers.TooltipHelper = {  
4   init: function() {  
5     return $(".ttip").tooltip({  
6       placement: "top"  
7     });  
8   }  
9 };  
10 ~  
11 jQuery(window.Helpers.TooltipHelper.init);  
12 ~  
13 $(document).on("page:load", window.Helpers.TooltipHelper.init);  
14 ~  
15 ~
```

```
1 window.Helpers ||= {}  
2 ~  
3 window.Helpers.TooltipHelper = {  
4   init: ->  
5     $(".ttip").tooltip placement: "top"  
6     # $(".mostrar").each ->  
7     # console.log $(this)  
8     # $(this).insertBefore "<div class='tooltip' title='Mostrar'><div cla  
9     # $(this).insertAfter "</div><div class='tooltip-arrow'></div></div>"  
10    # $(this).tooltip()  
11 }  
12 ~  
13 jQuery window.Helpers.TooltipHelper.init  
14 $(document).on "page:load", window.Helpers.TooltipHelper.init  
15 ~
```

Figura 10. Comparación entre sintaxis JavaScript y CoffeeScript

## 3.6. MYSQL

### 3.6.1. Mysql y la fusión con Ruby on Rails

#### 3.6.1.1. Introducción

Rails soporta la biblioteca SQLite por defecto. El acceso a la base de datos es totalmente abstracto desde el punto de vista del programador, es decir que es agnóstico a la base de datos, y Rails gestiona los accesos a la base de datos automáticamente (aunque, si se necesita, se pueden hacer consultas directas en SQL) Rails intenta mantener la neutralidad con respecto a la base de datos, la portabilidad

de la aplicación a diferentes sistemas de base de datos y la reutilización de bases de datos preexistentes.

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Aunque MySQL es software libre, MySQL AB distribuye una versión comercial de MySQL, que no se diferencia de la versión libre más que en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de no ser así, se vulneraría la licencia GPL.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración [14].

### **3.6.2. Características de Mysql**

La siguiente lista describe algunas de las características más importantes del software de base de datos MySQL:

1. Interioridades y portabilidad
2. Escrito en C y en C++
3. Probado con un amplio rango de compiladores diferentes
4. Funciona en diferentes plataformas.
5. Usa GNU Automake, Autoconf, y Libtool para portabilidad.
6. Uso completo de multi-threaded mediante threads del kernel. Pueden usarse fácilmente multipleCPUs si están disponibles.
7. Proporciona sistemas de almacenamiento transaccional y no transaccional.
8. Usa tablas en disco B-tree (MyISAM) muy rápidas con compresión de índice.
9. Relativamente sencillo de añadir otro sistema de almacenamiento. Esto es útil si desea añadir una interfaz SQL para una base de datos propia.
10. Un sistema de reserva de memoria muy rápido basado en threads [14].

### 3.6.2.1. Seguridad

Un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas está cifrado cuando se conecta con un servidor.

### 3.6.2.2. Escalabilidad y límites

Soporte a grandes bases de datos. Usamos MySQL Server con bases de datos que contienen 50 millones de registros. También conocemos a usuarios que usan MySQL Server con 60.000 tablas y cerca de 5.000.000.000.000 de registros.

### 3.6.2.3. Conectividad

En la figura 11 se muestra la conectividad que se realiza con Ruby on Rails en el sistema integrado web del Hospital Básico 7 BI Loja.

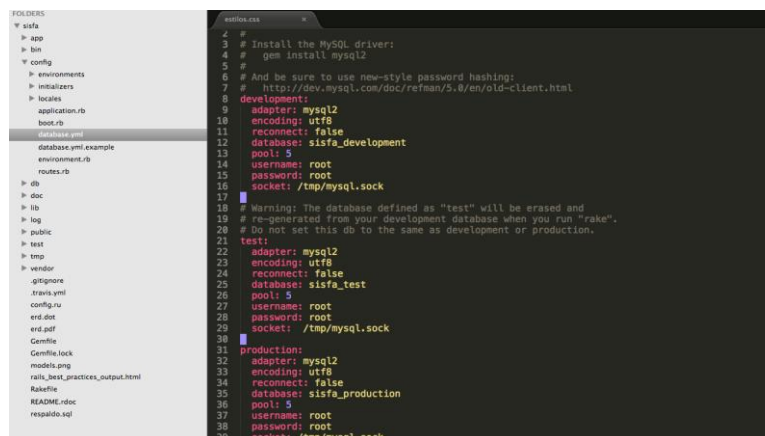


Figura 11. Conexión con la DB.



## **e. Materiales y Métodos**

### **1. Métodos**

El desarrollo del PFC requirió seguir los lineamientos de ciertos métodos, así como de técnicas e instrumentos que permitan la recopilación y análisis de la información necesaria para la presentación del PFC, tales como:

1. **Método Científico.-** Este método permitió conocer las funciones externas e internas de la entidad de salud, para llegar a determinar un problema y plantear una hipótesis general de los problemas que aquejan a esta institución.
  
1. **Método Inductivo.-** Se lo utilizó para determinar el problema general de investigación, por medio de la extracción de información que facilitaron las personas encargadas de los servicios antes mencionados, para llegar a la conclusión del problema global de esta entidad como es la falta perdida de información de los documentos físicos y su deterioro, la búsqueda de la información es muy tardía y no existe la comunicación adecuada entre los servicios, para la ágil atención al paciente.
  
1. **Método Deductivo.-** Este método permitió encontrar las soluciones adecuadas para los problemas específicos planteados en el presente PFC, ya que los problemas antes mencionados en los diferentes departamentos llevan a solucionar algo en común como es el desenvolvimiento correcto de los servicios de farmacia, central de enfermería y estadística, para así gestionar la información necesaria de una forma más rápida y oportuna.

1. **Método Bibliográfico.-** Este método ayudo a recolectar cantidad de información necesaria, precisa y contundente para el cumplimiento de los objetivos planteados, para así llegar a la solucionar el problema que se suscita hoy en día en el Hospital Básico 7 B.I Loja.

## 2. Técnicas

Las Técnicas que se utilizó para la recopilación de la información son los siguientes:

1. **Lectura comprensiva:** Consistió en obtener un conocimiento ordenado y sistemático de un aspecto de la realidad o de los acontecimientos hecho o ideas relacionadas con el tema específico.
1. **La Observación directa:** Esta técnica permitió apreciar los problemas encontrados respecto a la falta de comunicación entre departamentos y médicos y clientes.
1. **Entrevista:** Esta Técnica se utilizó para determinar problemas específicos ya que se la realizo con todas las personas encargadas de, Farmacia; que colaboraron con las respectivas preguntas planteadas por los investigadores para así llegar a formular y determinar el problema general del PFC.

### **3. Metodología de desarrollo de software**

Para el desarrollo del proyecto, se empleó la metodología XP (Programación Extrema), la cual tiene muchas ventajas como calidad, flexibilidad y un diseño simple. Además es una metodología ligera que trata de buscar métodos sencillos de obtener software de calidad.

Para suplir la falta de requisitos, casos de usos y demás herramientas utilizamos historias de usuarios tal como exige XP, además para el desarrollo del proyecto se debe considerar la implementación de módulos los mismos que deben contemplar las cuatro fases de la metodología XP (Planificación, Diseño, Desarrollo y Pruebas).

Además se cumple con las recomendaciones para emplear XP en este proyecto como son:

1. Interés sincero por la parte involucrada.
2. El equipo de trabajo pequeño, el cual consta de 1 persona.
3. Se dispone de una formación elevada y capacidad de aprender.

#### **Fases de la Metodología XP**

##### **Fase 1: Planificación del proyecto.**

**1. Historias De Usuario:** El primer paso es definir las historias de usuario con el cliente. Las cuales son usadas para estimar tiempos de desarrollo de la parte de la aplicación que describen. La descripción de estas se encuentra en el apartado 1.3.4. de la fase 1 de Resultados, son en total 16 historias de usuario, además existe un resumen de las mismas en el apartado 1.3.5.

**2. Plan de Entregas:** Después de tener ya definidas las historias de usuario es necesario crear un Release planing (plan de publicaciones), donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada versión del programa, el mismo se lo puede evidenciar en el apartado 1.3.6. De la fase 1 de Resultados.

**3. Iteraciones:** Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el "Release planning" que serán implementadas por iteraciones.

En el apartado 1.3.7. De la fase 1 de Resultados se encuentra la división de iteraciones planteada.

**4. Velocidad del proyecto:** Representa la rapidez con la que se desarrolla el proyecto; basta con contar el número de historias de usuario que se pueden implementar en una iteración, ver apartado 1.3.8. De la fase 1 de Resultados.

## **Fase 2: Diseño**

**Diseños simples:** Realizar los diseños de los módulos fácilmente entendibles, ver apartado 2.1 de la fase 2 de Resultados.

**Glosarios de términos:** Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código. Revisar apartado 2.2 de la fase 2 de Resultados.

**Refactorizar:** Supone revisar de nuevo estos códigos para procurar optimizar su funcionamiento.

**Tarjetas C.R.C.:** El uso de las tarjetas C.R.C (Class, Responsibilities and Collaboration) permiten centrarse y apreciar el desarrollo orientado a objetos. Las mismas que se pueden evidenciar en el apartado 2.3 de la fase 2 de Resultados.

### **Fase 3: Codificación**

Programar bajo estándares mantiene el código consistente y facilita su comprensión y escalabilidad.

En esta fase también encontramos las pruebas unitarias, las cuales están descritas en el apartado 3.2 de la fase 3 de Resultados.

### **Fase 4: Pruebas**

Se hace uso de test para comprobar el funcionamiento de los códigos que vayamos implementando.

En la fase 4 de Resultados se encuentra detallado todos los apartados aplicados. Se realizó una serie de pruebas como: pruebas de aceptación (4.2 de fase 4 de Resultados), pruebas funcionales (4.3.14 de fase 4 de Resultados), pruebas de Usabilidad (4.3.15 de fase 4 de Resultados), pruebas de seguridad (4.3.16 de fase 4 de Resultados) y pruebas de carga y estrés (4.3.17 de fase 4 de Resultados).

## **f. Resultados**

La presentación de resultados se encuentra dividida de acuerdo a las fases de la metodología XP.

### **FASE1. Planificación del Proyecto**

En esta primera fase se realizó un análisis de requisitos, con la finalidad de descubrir, modelar y delimitar los requisitos que han de satisfacer al sistema; es decir identificar de forma clara las funcionalidades que debe llevar a cabo el software.

#### **1.1 Introducción**

Esta sección recoge la Especificación de Requisitos Software (ERS), de la aplicación desarrollada, siguiendo el estándar IEEE 830-1998.

#### **1.2 Propósito**

Una de las fases más relevantes en el desarrollo de software es el levantamiento de requerimientos; a través de él, se permite entender mejor el problema, organizando y estructurando la información que el cliente haya brindado, se puede identificar las necesidades del negocio, los objetivos de la aplicación y los usuarios que tendrán interacción con ella.

Esta fase pretende orientar el trabajo inicial del desarrollo de software, dando descripciones preliminares de la aplicación, considerando algunos detalles de las secciones claves que la identifican.

Está dirigido a todas aquellas personas interesadas en el proyecto, principalmente al Hospital Básico 7 BI Loja y a las personas que hacen uso de sus servicios.

#### **1.3 Alcance**

El producto de este proyecto a desarrollar se identificara con el nombre “SISFA” se caracteriza por ser una aplicación web y multiplataforma. A continuación se presenta las especificaciones que el sistema debe cumplir.

Dentro de lo correspondiente a administraciones:

1. Administrar clientes: Ingresar, modificar, buscar. Los datos necesarios son: nombres, número de identificación, dirección, teléfono, correo.
2. Administrar Proveedores: Ingresar, modificar, buscar. Los datos necesarios son: nombres, número de identificación, Nombre de la empresa, razón social, dirección, teléfono, fax, ciudad, código, país.
3. Administrar Productos: Ingresar, modificar, buscar, ingresar lotes de productos, canje de producto: Los datos necesarios son: nombre, nombre genérico, código, categoría, casa comercial, precio de compra, precio de venta, ganancia, iva, fecha de caducidad, cantidad y lotes de productos.
4. Administrar Cuenta: Ingresar, Cambiar de clave, cambiar información de su cuenta. Los datos necesarios son: nombre de usuario, clave.
5. Administrar Usuarios: Ingresar, modificar clave, suspender cuenta, activar cuenta. Los datos necesarios son: nombre de usuario, clave, nombres, número de identificación.
6. Administrar Factura de ventas: Ingresar, anular, buscar. Los datos necesarios son: nombres y apellidos, número de identificación, teléfono, dirección, nombre de producto, cantidad, precio, iva, sub total 0%, subtotal 12% , iva 12%,total.
7. Administrar Factura de compras: Ingresar, buscar. Los datos necesarios son: nombres o razón social, número de identificación, teléfono, dirección, numero de factura, productos, cantidad, precio, iva, sub total 0%, subtotal 12% , iva 12%,total.
8. Administrar comprobantes de transferencias: Ingresar, anular, buscar. Los datos necesarios son: nombre de receptor, nombre de emisor, dependencia, nombre de producto, cantidad, valor unitario, iva, total.
9. Facturar receta: facturar. Los datos necesarios son: nombres y apellidos, número de identificación, teléfono, dirección, nombre de producto, cantidad, precio, iva, sub total 0%, subtotal 12% , iva 12%,total.

El sistema también:

1. Registra cierre de caja diarios de venta, y transferencia.
2. Permite realizar reportes de hospitalización de un paciente, para determinar el consumo de medicamento que tuvo este durante los días de hospitalización.
3. Permite ver cierres de caja mensuales de venta, hospitalización y transferencia.
4. Permite realizar reportes de venta, hospitalización, transferencia y compras, con fecha de inicio y fecha final, con su respectiva impresión.
5. Permite realizar reportes de cierre de caja diario y mensual, con su respectiva impresión.
6. Permite generar reportes de liquidaciones mensuales y con su respectiva impresión.
7. Permite realizar el inventario general por casa comercial de los productos.
8. Permite generar un kardex de cada producto para el control de entrada y salida de producto.
9. Controlar las alertas de caducidad del medicamento con meses de anticipación.
10. Realiza el respectivo canje de producto que esta al caducar

El objetivo es “Desarrollar un Sistema Web de gestión de farmacia, basado en tecnología móvil para el Hospital Básico 7 B.I Loja”, con el fin que la institución cuente con un software que ayude al mejor desenvolvimiento y eficacia en sus labores, brindando al personal una herramienta fácil de usar, buscando como metas:



1. Facilitar al usuario que la información la tenga siempre disponible.
2. Brindar rapidez en el control de productos.
3. Mejorar el desenvolvimiento laboral del personal.
4. Mejorar la calidad de servicio a la ciudadanía en general.

#### **1.4 Definiciones, siglas y abreviaciones.**

1. Usuario: Persona(s) que hace uso del software, en la aplicación se encuentran diferentes tipos de usuario como: Súper Administrador, Administrador Farmacia, vendedor, visitador.
1. Súper Administrador: Usuario único que puede crear y asignar cuentas.
2. Administrador Farmacia: Usuario que se encuentra a cargo de todo el departamento de Farmacia.
3. Vendedor: Usuario que solo se encarga de emitir ventas, transferencias, anular y despachar reportes de hospitalización.
4. Visitador: persona que solo puede consultar la parte de reportes financieros.
5. Cliente: Persona que ha realizado una compra en farmacia.
6. Factura: Comprobante de que se ha realizado una compra en esta dependencia.
7. Proforma: Comprobante que se da a un cliente para la consulta de medicamento con sus respectivos precios.
8. Transferencia: Comprobante que se emite por la entrega de medicamento a otras dependencias ejemplo quirófano, rayos x, etc.
9. Hospitalización: Comprobante de consumo de medicamento de un paciente que se encuentre hospitalizado.
10. Proveedor: Persona O Empresa que provee de productos a esta institución.
11. Canje: proceso que se realiza en el caso que el producto este al caducar y solo realiza por medio del proveedor.
12. Kardex: Informe de movimiento de producto ya sea por medio de venta, transferencia u hospitalización.

13. Inventario: Lista de producto que se encuentra disponible en farmacia.
14. Liquidación: Informe económico de gastos mensuales y venta mensual para dar a conocer la utilidad o ganancia de esta institución.
15. Cierre de caja diario: informe de la venta diaria que se hizo.
16. Cierre de caja mensual: informe de la venta, hospitalización y transferencia de productos durante el mes.
17. Producto: Elemento que ingresa a formar parte del inventario para poder ser comercializado.

## **2. Descripción global**

### **2.1. Perspectiva del producto**

Este proyecto de ingeniería de software está basado en lenguajes web, por lo cual la arquitectura es multiplataforma, siendo independiente del sistema operativo, por tal motivo no es necesario establecer relación con otros sistemas.

El diseño de la interfaz “UI” se ha realizado de manera clara y sencilla, la cual es intuitiva permitiendo que no sea confusa en la parte de Experiencia Usuario “UX”.

### **2.2. Funciones del producto**

A continuación se enmarca las funciones que cumple el módulo:

Farmacia:

- Administrar la información de los vendedores.
- Suspensión de vendedores.
- Administrar la información de los productos.
- Administrar la información de productos caducados.
- Administración de clientes.
- Administración de proveedores.

- Cierres de caja diarios de la venta del día.
- Facturar recetas emitidas por el médico.
- Facturación de venta de producto.
- Facturación de compra de producto.
- Transferencia de producto a otra dependencia del hospital.
- Administración de inventario.
- Despacho de producto cuando hay una hospitalización de paciente.
- Canjes de producto caducado.
- Generación automática de reportes diarios, mensuales de productos vendidos, transferidos u hospitalización.
- Generación automática de kardex de cualquier producto.
- Visualización de facturas de venta que se hizo durante los transcurso de los días.
- Informe mensual de producto que queda.
- Creación de reportes de liquidación mensual.

Para describir de forma más precisa las funcionalidades identificadas para el Sistema, se han organizado de la siguiente manera, la TABLA I, que describe los perfiles de usuarios de la aplicación:

TABLA I : ACTORES DEL SISTEMA

ACTORES DEL SISTEMA	FUNCIONES
Administrador	<ul style="list-style-type: none"> <li>• Crear usuario administrador de Farmacia</li> <li>• Crear usuario Vendedor.</li> <li>• Suspende usuarios</li> </ul>
Administrador de Farmacia	<ul style="list-style-type: none"> <li>• Registrar vendedores</li> <li>• Suspende vendedores</li> <li>• Administrar producto</li> <li>• Canjear Producto</li> <li>• Administrar Clientes</li> <li>• Administrar Proveedor</li> <li>• Facturar recetas</li> <li>• Cerrar caja diaria</li> <li>• Facturas de venta</li> <li>• Facturas de compra</li> <li>• Transferencias de producto</li> <li>• Despachar producto de hospitalizado</li> <li>• Reportes mensuales de venta</li> <li>• Reportes mensuales de Hospitalizados</li> <li>• Reportes mensuales de transferencias</li> <li>• Reportes mensuales de liquidación.</li> <li>• Reportes mensuales de inventario</li> </ul>
Vendedor	<ul style="list-style-type: none"> <li>• Facturas de venta</li> <li>• Cierre de caja</li> <li>• Administrar clientes</li> <li>• Emitir proformas</li> </ul>

### **2.3. Características del Usuario**

Los usuarios del sistema son los funcionarios que trabajan en los diferentes departamentos del hospital, los cuales están categorizados como:

Administrador: Jefe del departamento de sistemas del hospital.

Administrador Farmacia: responsable del departamento de Farmacia.

Vendedor: Ayudante de venta de producto.

Dependiendo de su perfil, el usuario podrá ejercer las funciones descritas en el apartado 2.2 TABLA I actores del sistema.

### **2.4. Restricciones**

Debido a que se ha hecho uso de nuevas tecnologías como CSS3 y JS, el navegador debe tener habilitado el uso de JS, pero el mismo sufrirá ciertos cambios en la parte visual, al abrirlo en versiones inferiores a IE8, pero no existe ningún problema en cuanto a las operaciones.

### **2.5. Atención y dependencias**

Algunos factores que pueden afectar los requerimientos del sistema son:

1. Agregar nuevos roles de usuarios los cuales ya están definidos.
2. Cambiar el diseño de los reportes impresos.
3. Modificar el curso de los eventos en las acciones de los usuarios con el sistema.

### **2.6. Prorratear los requisitos**

Algunas mejoras que pueden plantearse para un futuro son:

1. Agregar nuevos reportes de administración de productos,
2. Permitir al sistema conectarse con el sistema del ejercito ISSFA, para validar y autocompletar la información de pacientes militares.
3. Consulta de medicamento por internet.

### 3. Los requisitos específicos

#### 3.1. Requerimientos Funcionales

El sistema permitirá a todos los usuarios:

TABLA II : PERMISOS DE USUARIOS.

REF.	DESCRIPCIÓN	CATEGORÍA
RF-1.01	Acceder al sistema mediante el inicio de sesión con la clave correspondiente.	Oculto
RF-1.02	Cambiar su clave.	Evidente

El sistema permitirá al administrador:

TABLA III : PERMISOS DE ADMINISTRADOR.

REF.	DESCRIPCIÓN	CATEGORÍA
RF-1.03	Anular cuentas de usuarios	Evidente
RF-1.04	Acceso a todas las funciones de Farmacia	Oculto

El sistema permitirá al administrador Farmacia:

TABLA IV : PERMISOS DE ADMINISTRADOR DE FARMACIA.

REF.	DESCRIPCIÓN	CATEGORÍA
RF-1.05	Administrar producto(Ingresar, modificar, ingresar lotes de productos, canje de producto)	Evidente
RF-1.06	Administrar clientes (Ingresar, modificar)	Evidente
RF-1.07	Administrar Proveedores: (Ingresar, modificar)	Evidente
RF-1.08	Administrar Cuenta: (Ingresar, Cambiar de clave, cambiar información de su cuenta)	Evidente
RF-1.09	Administrar Usuarios: (Ingresar, modificar clave, suspender cuenta, activar cuenta)	Evidente
RF-1.10	Administrar Factura de ventas: (Ingresar, anular)	Evidente
RF-1.11	Administrar Factura de compras: (Ingresar producto)	Evidente
RF-1.12	Administrar comprobantes de transferencias: Ingresar, anular.	Evidente
RF-1.13	Facturar rectas: facturar.	Evidente
RF-1.14	Cierres de caja diarios	Evidente
RF-1.15	Administración de inventario	Evidente
RF-1.16	Despacho de producto	Evidente
RF-1.17	Generar reportes diarios, mensuales de productos vendidos	Evidente
RF-1.18	Generar kardex de cualquier producto	Evidente
RF-1.19	Visualización de facturas de venta durante el día	Evidente
RF-1.20	Visualización de inventario restante del mes	Evidente
RF-1.21	Reportes de liquidación mensual	Evidente

El sistema permite al vendedor:

TABLA V: PERMISOS DE VENDEDOR.

REF.	DESCRIPCIÓN	CATEGORÍA
RF-1.22	Administrar Factura de ventas: (Ingresar)	Evidente
RF-1.17	Cierre de caja	Evidente
RF-1.18	Administrar clientes (Ingresar, modificar)	Evidente
RF-1.19	Facturar recetas: facturar.	Evidente

### 3.2. Requerimientos No Funcionales

TABLA VI: REQUERIMIENTOS NO FUNCIONALES.

REF.	DESCRIPCIÓN
RNF-01	El sistema deberá validar número de identificación.
RNF-02	El sistema deberá mostrar notificaciones ante errores.
RNF-03	El sistema deberá pedir confirmación antes de eliminar algún registro, factura u otro elemento.
RNF-04	El sistema deberá ser intuitivo.
RNF-05	El sistema deberá brindar tiempos de repuestas accesibles en las Operaciones
RNF-06	El sistema deberá mostrar etiquetas claras para campos, botones, menús.

### 3.3. Equipo: Integrantes y Roles

TABLA VII : MIEMBROS DEL PROYECTO.

Miembro	Roles XP	Metodología
Ing. Marco Ocampo	Directrto de tesis encargado de seguimiento	<b>XP</b>
Angel Valdez	Programador, encargado del módulo de farmacia y pruebas	
Lenin Capa	Programador, encargado del módulo de Estadística doctores y enfermería.	



### 3.4. Historias de Usuarios

TABLA VIII: HISTORIA DE USUARIO: ADMINISTRACIÓN USUARIOS

Historia de Usuario	
<b>Número:</b> 1	<b>Nombre:</b> Administración usuarios
<b>Usuario:</b> administrador	
<b>Iteración asignada:</b> 1	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 1.5
<b>Descripción:</b> <p>El administrador puede realizar ingresos de nuevos usuarios, modificaciones, anular o activar. Los tipos de usuario son: Administrador, Administrador estadística, Administrador_enfermería, Administrador_farmacia, Auxiliar_estadística, vendedor, enfermera, doctor.</p>	
<b>Observaciones:</b> <div>  <div>   </div> </div>	

TABLA IX : HISTORIA DE USUARIO ADMINISTRACIÓN CLIENTES



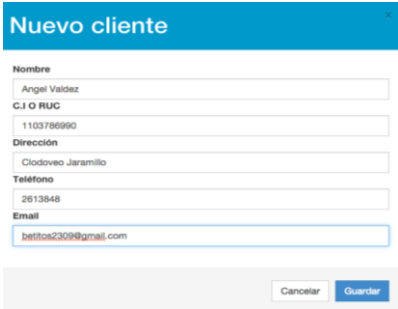

Historia de Usuario	
Número: 2	Nombre: Administración de clientes
Usuario: Administrador farmacia y vendedor	
Iteración asignada: 2	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 2
<b>Descripción:</b> El administrador de farmacia y vendedor pueden crear clientes, así como también realizar modificaciones y actualizar.	
<b>Observaciones:</b> <div>        </div>	

TABLA X : HISTORIA DE USUARIO: ADMINISTRAR PRODUCTO

<b>Historia de Usuario</b>	
<b>Número:</b> 3	<b>Nombre:</b> Administrar producto
<b>Usuario:</b> Administrador farmacia	
<b>Iteración asignada:</b> 2	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.5
<b>Riesgo en Desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 1.5
<b>Descripción:</b> El administrador de farmacia puede ingresar producto	
<b>Observaciones:</b>	

Factura de Compra
27 de septiembre de 2014

**Datos del Proveedor**

  
proveedor

**Productos**

Subtotal12%:

Subtotal 0%:

Iva 12%:

TOTAL:

Guardar

Factura de Compra
27 de septiembre de 2014

**Datos del Proveedor**

**Productos**

x finalin \$96.00
↑

Precio de compra

Ganancia

Tiene IVA?  
☐

Fecha Caducidad

Cantidad

Lote

+ Compra

+ Item

Subtotal12%:

Subtotal 0%:

Iva 12%:

TOTAL:

Guardar

TABLA XI: HISTORIA DE USUARIO: ADMINISTAR PROVEEDORES



Historia de Usuario	
<b>Número:</b> 4	<b>Nombre:</b> Administración de proveedores
<b>Usuario:</b> Administrador farmacia	
<b>Iteración asignada:</b> 2	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.5
<b>Riesgo en Desarrollo:</b> Alto (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 1
<b>Descripción:</b> El administrador de farmacia puede ingresar a un proveedor cuando trae producto nuevo	
<b>Observaciones:</b> <div data-bbox="585 1052 1248 1335">  </div> <div data-bbox="729 1379 1080 1861">  </div>	

TABLA XII : HISTORIA DE USUARIO: ADMINISTRAR FACTURA DE VENTA


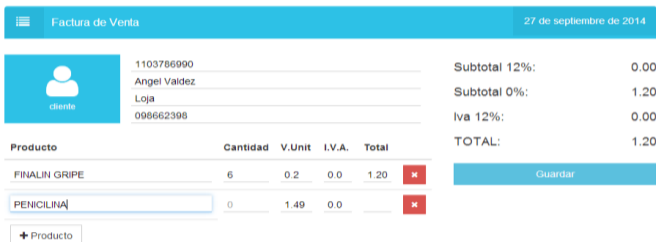
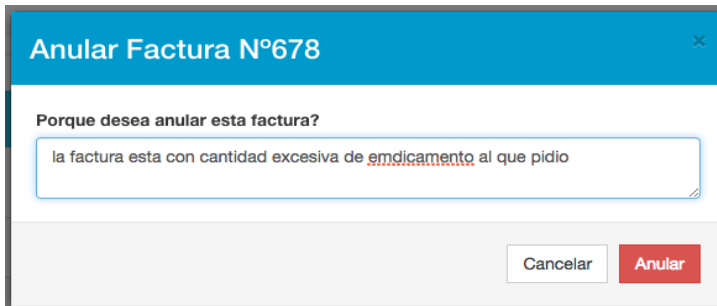
Historia de Usuario	
Número: 5	Nombre: Factura de venta
Usuario: Administrador de farmacia y vendedor	
Iteración asignada: 3	
Prioridad en Negocio: Media (Alta / Media / Baja)	Puntos Estimados: 1
Riesgo en Desarrollo: Media (Alto / Medio / Bajo)	Puntos Estimados: 1.5
<b>Descripción:</b> El administrador de farmacia y el vendedor pueden otorgar facturas diarias de venta de producto, y además anular facturas por cualquier situación ajena al usuario.	
<b>Observaciones:</b> <div>     </div>	
<b>Anular factura:</b> <div>  </div>	

TABLA XIII : HISTORIA DE USUARIO ADMINISTRAR FACTURA DE COMPRA

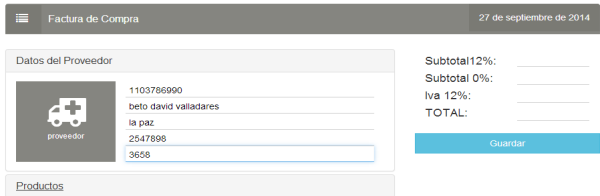
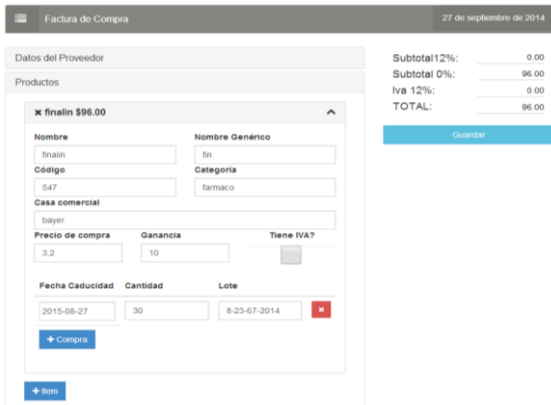
Historia de Usuario	
<b>Número:</b> 6	<b>Nombre:</b> Administración de usuarios
<b>Usuario:</b> Administrador farmacia	
<b>Iteración asignada:</b> 3	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 1.5
<b>Riesgo en Desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 2
<b>Descripción:</b> El Administrador de farmacia puede: crear, editar, eliminar y suspender vendedores.	
<b>Observaciones:</b> <div>  </div> <div>  </div>	

TABLA XIV : HISTORIA DE USUARIO: ADMINISTRAR COMPROBANTES DE TRANSFERENCIAS DE PRODUCTO

Historia de Usuario	
<b>Número:</b> 7	<b>Nombre:</b> transferencias de producto
<b>Usuario:</b> Administrador de farmacia y vendedor	
<b>Iteración asignada:</b> 3	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.5
<b>Riesgo en Desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 1.5
<b>Descripción:</b> El Administrador de farmacia y vendedor pueden: Transferir producto a las dependencias que tiene el hospital es similar a una factura, como también anular la transferencia por cualquier devolución de producto.	
<b>Observaciones:</b> <div>  </div>	
<b>Anular transferencia:</b> <div>  </div>	

TABLA XV : HISTORIA DE USUARIO: ADMINISTRAR RECETAS

Historia de Usuario

Número: 8

Nombre: Realizar proforma

Usuario: Administrador farmacia y vendedor

Iteración asignada: 4

Prioridad en Negocio: Media  
(Alta / Media / Baja)

Puntos Estimados: 0.5

Riesgo en Desarrollo: Media  
(Alto / Medio / Bajo)

Puntos Estimados: 1.5

Descripción:

El Administrador de farmacia y el vendedor pueden facturar las recetas emitidas por el doctor.

Observaciones:

RECETAS EMITIDAS

Buscar:

Cliente	Número	Total	Mostrar
Amanda Diaz Orozco	8	1.83	
Judith Valdez	7	2.42	

Mostrando registros del 1 al 2 de un total de 2 registros (filtrado de un total de 7 registros)

Receta N°8

Cliente: Amanda Diaz Orozco

Fecha de Emisión: 2014-11-05

Producto	Cantidad	Precio	Iva	Total
ALCHOL ETILICO 200ML	1.0	0.61	0.0	0.61
buprex	3.0	0.37	0.0	1.11
GASA	1.0	0.11	0.0	0.11

Sub 12%: \$ 0.0

Sub 0%: \$ 1.83

IVA: \$ 0.0

Total: \$ 1.83

Salir

Facturar

Facturar



TABLA XVI : HISTORIA DE USUARIO: CIERRE DE CAJA


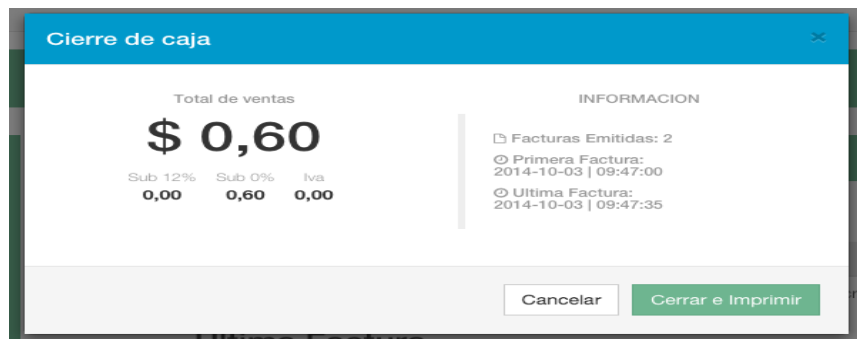
Historia de Usuario	
<b>Número:</b> 9	<b>Nombre:</b> Cierre de caja
<b>Usuario:</b> Administrador de farmacia y vendedor	
<b>Iteración asignada:</b> 4	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 1.5
<b>Descripción:</b> El Administrador de farmacia y el vendedor pueden cerrar caja una vez terminado su horario de trabajo para así comprobar sus estados de venta del día.	
<b>Observaciones:</b>  	

TABLA XVII : HISTORIA DE USUARIO: ADMINISTRACIÓN DE INVENTARIO

Historia de Usuario

Número: 10

Nombre: Reporte inventario

Usuario: Administrador farmacia

Iteración asignada: 3

Prioridad en Negocio: Media  
(Alta / Media / Baja)

Puntos Estimados: 0.5

Riesgo en Desarrollo: Alta  
(Alto / Medio / Bajo)

Puntos Estimados 1.5

Descripción:  
El Administrador farmacia puede ver cuánto producto existe en stock y así saber que falta por adquirir, también como saber quién vendió el producto.

Observaciones:

← INVENTARIO DE PRODUCTOS

Descargar: PDF

Producto	Cantidad	Precio C	Total
farmaalemana			
buprex	475	\$0.34	\$161.5
FARM-EC			
GUANTES QUIRURGICOS	172	\$0.55	\$94.6
ALGODON 500GR	191	\$0.2	\$38.2
ALCHOL ETILICO 200ML	45	\$0.55	\$24.75
GASA	94	\$0.1	\$9.4
FARMEI			
FINALIN GRIPE	139	\$0.18	\$25.02
JHONSON			
ACEITE JHONSON	98	\$2.2	\$215.6
MEDISUMI			
FLUCONASOL 200ml	100	\$4.0	\$400.0
TOTAL:			\$984,69

TABLA XVIII : HISTORIA DE USUARIO: DESPACHO DE PRODUCTO

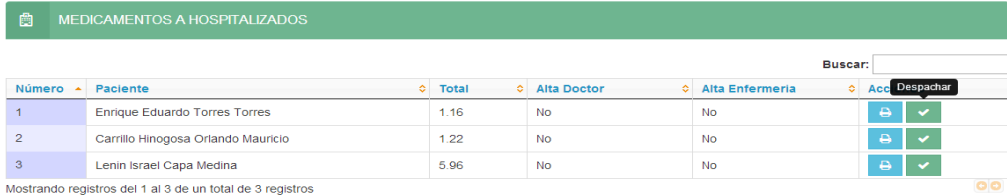

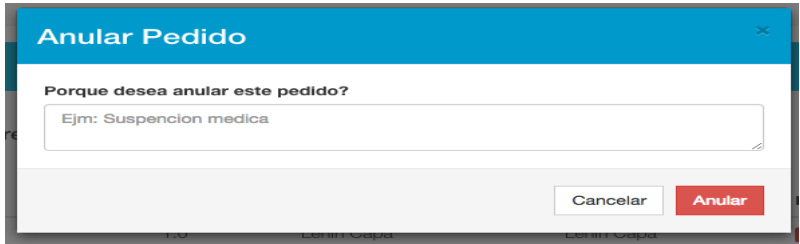
Historia de Usuario	
<b>Número:</b> 11	<b>Nombre:</b> Despacho de producto de un hospitalizado
<b>Usuario:</b> Administrador farmacia y vendedor	
<b>Iteración asignada:</b> 3	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.5
<b>Riesgo en Desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 1.5
<b>Descripción:</b> El Administrador de farmacia y el vendedor pueden despachar el producto que desde central de enfermería piden para poder suministrar al paciente.	
<b>Observaciones:</b>  <div>  </div> <div>  </div>	
<b>Anular pedido de hospitalización:</b>  <div>  </div>	

TABLA XIX : HISTORIA DE USUARIO: GENERAR REPORTES

Historia de Usuario

Número: 12

Nombre: Reportes de ventas

Usuario: Administrador farmacia

Iteración asignada: 4

Prioridad en Negocio: Media

(Alta / Media / Baja)

Puntos Estimados: 0.5

Riesgo en Desarrollo: Alta

(Alto / Medio / Bajo)

Puntos Estimados: 1.5

Descripción:

El Administrador de farmacia puede generar los reportes de ventas del día, transferencias o despacho por hospitalización.

Observaciones:

Reportes

Ventas y Compras

Hospitalización

Ventanilla

Hospitalización

Transferencias

Compra

Imprimir

Cierre de Caja

Ingrese Fecha (mes):

Fecha

Ir

Ingrese Fecha (dia):

Fecha

Ir

Liquidaciones

Ingrese Mes:

Fecha

Ir

Reporte de Facturas de Ventanilla desde:

2014-09-09 hasta 2014-09-30

Descargar: PDF

Nº	Cliente	Dirección	Vendedor	Fecha	Sub 12%	Sub 0%	IVA	Total
685	Flor Maria Collaguazo Narváez	Clodoveo	angel	2014-09-24   11:08:48	0.0	51.06	0.0	51.06
686	Fabrizio Flores	Loja	judith	2014-09-27   11:44:42	0.0	1.0	0.0	1.0
					0,00	52,06	0,00	52,06

Facturas Emitidas: 2

TABLA XX : HISTORIA DE USUARIO: GENERAR KARDEX DE PRODUCTO

Historia de Usuario	
<b>Número:</b> 13	<b>Nombre:</b> Reporte de kardex
<b>Usuario:</b> Administrador farmacia	
<b>Iteración asignada:</b> 3	
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.5
<b>Riesgo en Desarrollo:</b> Media (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 1.5
<b>Descripción:</b> El Administrador de farmacia puede ver el kardex de todo los productos que existen en el inventario, consta de saber qué fecha salió e ingresa producto en farmacia	
<b>Observaciones:</b> <div> <div> <div>LISTA DE PRODUCTOS</div> <div> <div>Inventario</div> <div>Importar productos</div> </div> <div> <div> <div>Nombre</div> <div>Código</div> <div>Casa Comercial</div> <div>Kardex</div> </div> <div> <div>ACEITE JHONSON</div> <div>JHAC</div> <div>JHONSON</div> <div> <div>←</div> <div>→</div> <div>+</div> <div>-</div> </div> </div> <div> <div>ALCHOL ETILICO 200ML</div> <div>ALCHET2</div> <div>FARM-EC</div> <div> <div>←</div> <div>→</div> <div>+</div> <div>-</div> </div> </div> <div> <div>ALGODON 500GR</div> <div>ALGD</div> <div>FARM-EC</div> <div> <div>←</div> <div>→</div> <div>+</div> <div>-</div> </div> </div> <div> <div>FINALIN GRIPE</div> <div>FNIL</div> <div>FARMEI</div> <div> <div>←</div> <div>→</div> <div>+</div> <div>-</div> </div> </div> <div> <div>GASA</div> <div>GSA</div> <div>FARM-EC</div> <div> <div>←</div> <div>→</div> <div>+</div> <div>-</div> </div> </div> <div> <div>GUANTES QUIRURGICOS</div> <div>GTSQ</div> <div>FARM-EC</div> <div> <div>←</div> <div>→</div> <div>+</div> <div>-</div> </div> </div> <div> <div>PENICILINA</div> <div>PNCL</div> <div>BAYER</div> <div> <div>←</div> <div>→</div> <div>+</div> <div>-</div> </div> </div> </div> <div>Mostrando registros del 1 al 7 de un total de 7 registros</div> </div> <div> <div>Kardex de ALCHOL ETILICO 200ML</div> <div> <div>Stock</div> <div>Inversion</div> <div>Dinero Venta</div> </div> <div> <div>44</div> <div>\$24.2</div> <div>\$0.0</div> </div> <div> <div> <div>Fecha</div> <div>Tipo</div> <div>Módulo</div> <div>Cantidad</div> <div>Precio</div> <div>Observaciones</div> </div> <div> <div>2014-08-08</div> <div>Entrada</div> <div></div> <div>50</div> <div>0.55</div> <div></div> </div> <div> <div>2014-08-08</div> <div>Salida</div> <div>Hospitalizacion</div> <div>1</div> <div>0.61</div> <div></div> </div> <div> <div>2014-08-09</div> <div>Salida</div> <div>Ventanilla</div> <div>1</div> <div>0.61</div> <div></div> </div> <div> <div>2014-08-11</div> <div>Salida</div> <div>Ventanilla</div> <div>1</div> <div>0.61</div> <div></div> </div> <div> <div>2014-08-11</div> <div>Salida</div> <div>Ventanilla</div> <div>1</div> <div>0.61</div> <div></div> </div> <div> <div>2014-08-24</div> <div>Salida</div> <div>Hospitalizacion</div> <div>2</div> <div>0.61</div> <div></div> </div> </div> <div>Mostrando registros del 1 al 6 de un total de 6 registros</div> </div> </div>	

TABLA XXI : HISTORIA DE USUARIO: REPORTES DE FACTURAS DEL DÍA

Historia de Usuario

Número: 14

Nombre: Ver facturas del día

Usuario: Administrador de farmacia

Iteración asignada: 4

Prioridad en Negocio: Alta

(Alta / Media / Baja)

Puntos Estimados: 1

Riesgo en Desarrollo: Alta

(Alto / Medio / Bajo)

Puntos Estimados: 1.5

Descripción:

El Administrador de farmacia puede ver las facturas emitidas durante todo el día de los vendedores, así saber cuántas facturas se emitieron y cuantas se anulan también durante el transcurso del turno del vendedor.

Observaciones:

FACTURAS EMITIDAS

Venta

Anuladas

Compra

Buscar:

Cliente	Nº Factura	Total	Vendedor	Accio	Anular
Fabricio Flores	678	1.24	Ienin		
Fabricio Flores	686	1	Judith		
Angel Valdez	679	1.69	Ienin		
Daniel Francisco Reinoso Ortega	681	13.62	Ienin		
Daniel Francisco Reinoso Ortega	682	0.72	Ienin		
Daniel Francisco Reinoso Ortega	683	0.72	Ienin		
Daniel Francisco Reinoso Ortega	684	7.45	angel		
Gonzalez Cano Edwin Alberto	680	3.38	Ienin		
Flor Maria Collaguazo Narváez	685	51.06	angel		

Mostrando registros del 1 al 9 de un total de 9 registros

TABLA XXII : HISTORIA DE USUARIO: REPORTE DE INVENTARIO MENSUAL

Historia de Usuario

Número: 15

Nombre: Ver Inventario

Usuario: Administrador de farmacia

Iteración asignada: 4

Prioridad en Negocio: Media  
(Alta / Media / Baja)

Puntos Estimados: 1

Riesgo en Desarrollo: Media  
(Alto / Medio / Bajo)

Puntos Estimados: 1.5

Descripción:  
El Administrador de farmacia puede: ver el inventario de producto mensual ver el total de dinero por producto y el total de dinero que queda para empezar el siguiente mes

Observaciones:

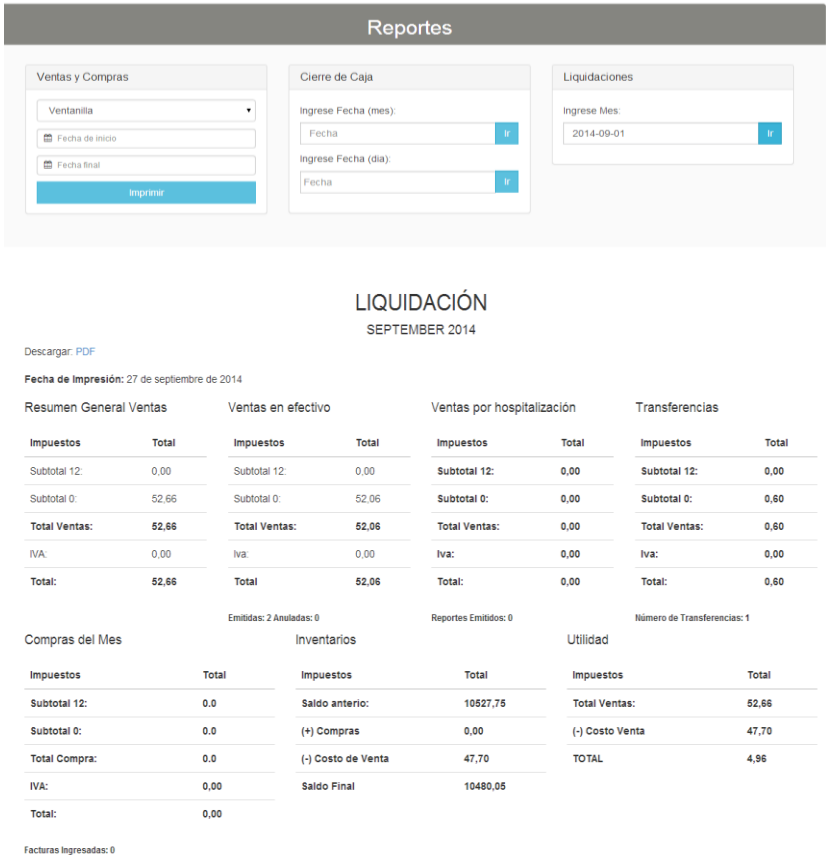
SISFA

angel

Inventario

Casa comercial	Producto	Cantidad	Precio Compra	Inversion
BAYER	PENICILINA	69	\$ 1.35	\$ 93.15
FARM-EC	GUANTES QUIRURGICOS	178	\$ 0.55	\$ 97.9
FARMEI	FINALIN GRIPE	181	\$ 0.18	\$ 32.58
FARM-EC	ALGODON 500GR	196	\$ 0.2	\$ 39.2
FARM-EC	ALCHOL ETILICO 200ML	44	\$ 0.55	\$ 24.2
FARM-EC	GASA	93	\$ 0.1	\$ 9.3
JHONSON	ACEITE JHONSON	98	\$ 2.2	\$ 215.6

TABLA XXIII : HISTORIA DE USUARIO: REPORTES DE LIQUIDACIÓN MENSUAL

Historia de Usuario	
<b>Número:</b> 16	<b>Nombre:</b> Liquidación mensual
<b>Usuario:</b> Administrador de farmacia	
<b>Iteración asignada:</b> 4	
<b>Prioridad en Negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos Estimados:</b> 2
<b>Riesgo en Desarrollo:</b> Alta (Alto / Medio / Bajo)	<b>Puntos Estimados:</b> 2
<b>Descripción:</b> El Administrador de farmacia saca su reporte de liquidación mensual donde consta lo vendido, lo comprado, lo transferido y lo de hospitalización, y al final un reporte de ganancia mensual que adquirió durante ese mes.	
<b>Observaciones:</b> <div style="text-align: center;">  </div>	



### 3.5.Resumen de Historias de Usuario

TABLA XXIV : RESUMEN DE HISTORIAS DE USUARIO

<b>N</b>	<b>Nombre</b>	<b>Prioridad</b>	<b>Riesgo</b>	<b>Iteración</b>
1	Administrar_usuarios	Media	Media	1
2	Administración de Clientes	Media	Alta	2
3	Administrar producto	Media	Alta	2
4	Administar proveedores	Media	Alta	2
5	Administar Facturas de venta	Media	Media	1
6	Administrar factura de compra	Media	Alta	1
7	Administración de comprobantes de transferencias de producto	Media	Media	1
8	Administrar recetas	Media	Media	2
9	Cierre de caja	Media	Alta	2
10	Administración de inventario	Media	Alta	1
11	Despacho de producto	Media	Alta	2
12	Generar reportes	Media	Alta	1
13	Generar kardex de producto	Media	Media	2
14	Reportes de factura del día	Alta	Alta	2
15	Reportes inventario mensual	Media	Media	1
16	Reportes de liquidación mensual	Alta	Alta	2

### 3.6. Plan de Entregas

TABLA XXV: PLAN DE ENTREGAS.

N	HISTORIA DE USUARIO	TIEMPO ESTIMADO (CALENDARIO ESTIMADO)			ITERACIÓN ASIGNADA			
		SEMANAS ESTIMADAS	DÍAS ESTIMADOS	HORAS ESTIMADAS	1	2	3	4
1	Administración de usuarios	3	15	45	X			
2	Administrar clientes	3	15	45		X		
3	Administrar Producto	4	20	110		X		
4	Administrar proveedores	3	15	45		X		
5	Administrar Factura de venta	3	15	60			X	
6	Administrar factura de compra	3	15	60			X	
7	Administrar comprobantes de transferencias de producto	2	10	40			X	
8	Administrar recetas	2	10	20				X
9	Cierre de caja	2	10	60				X
10	Administración de inventario	2	10	30			X	
11	Despacho de producto	2	10	30			X	
12	Generar reportes	1	5	25				X
13	Generar Kardex de producto	1	5	20			X	
14	Reportes de facturas del día	1	5	25				X
15	Reportes de inventario mensual	1	5	25				X
16	Reportes de liquidación mensual	1	5	25				X
TIEMPO TOTAL ESTIMADO		34	170	665				

### **3.7. División en iteraciones.**

La tesis está contemplada en cuatro iteraciones, por consiguiente con un total de cuatro entregas, por lo cual se desarrolló partes de la aplicación completamente funcionales.

En la primera iteración se refiere al módulo de control de usuarios, consta de los siguientes partes: creación de roles, permisos, suspensión de usuarios, en la segunda iteración la administración de clientes, productos y proveedores la cual consta de : crear clientes, modificar clientes, crear producto, modificar producto, canjear producto, crear proveedores, modificar proveedores, en la tercera iteración se realizó facturas de venta, facturas de compra, transferencias, y hospitalización que constan de : facturar producto, anular facturas, ver lista de facturas, ingresar facturas de compra, ver facturas de compra, transferir producto a las diferentes dependencias, anular transferencias, ver transferencias, despachar producto que pidió central de enfermería o quirófano, anular pedidos de hospitalización o quirófano y en la cuarta iteración se realizó reportes financieros, cierres de caja y despacho de recetas que son: reportes de ventanillas, reportes de transferencias, reportes de hospitalizaciones, reportes mensuales de cierre de caja, y reportes mensuales de liquidación que comprende al final la utilidad mensual o ganancia que se tiene en el mes, cierre de caja diario que lo realiza el usuario que se encuentre vendiendo durante su jornada de trabajo, el despacho de recetas se lo hace siempre y cuando el médico haya recetado medicamento a un paciente que fue por consulta externa.

El periodo tomado para la planeación de la primera iteración fue tres semanas, debido a que solo fue creación de usuarios y sus respectivos roles como se explica en las historias de usuario, y así definir la interfaz que saldrá en cada una de las funciones antes mencionadas, en la segunda iteración se tomó diez semanas para definir la administración de productos, proveedores, clientes ya que son un poco más extensos los métodos y validaciones que se dio, así como también las relaciones estén acorde en la base de datos, la tercera iteración comprendió diez y siete semanas donde se realizó los procesos correspondientes a este departamento y la cuarta iteración comprendió cuatro semanas donde se da a conocer ya en si el producto al ciento por ciento.

### 3.8. Velocidad del proyecto

TABLA XXVI : VELOCIDAD DEL PROYECTO

	Iteración 1	Iteración 2	Iteración 3	Iteración 4
<b>Horas</b>	45	200	240	180
<b>Semanas</b>	3	10	13	8
<b>Horas semanales</b>	15	20	14	45
<b>Historias de usuario (velocidad del proyecto)</b>	1	3	6	6

### 3.9. Entregas Pequeñas

Se procedió a liberar nuevas versiones según la funcionalidad que se fue implementando en el proyecto:

TABLA XXVII : ITERACIONES

Iteración	Fecha de entrega
1ra	11 de abril de 2014
2da	20 de junio de 2014
3ra	12 de septiembre de 2014
4ta	14 de noviembre 2014

## 2. FASE 2.- Diseño

Con la metodología XP el diseño se realiza durante todo el tiempo de vida del proyecto, siendo constantemente revisado y muy probablemente modificado debido a cambios presentados durante el desarrollo.

## 2.1. Diseños Simples

En lo que respecta a la interfaz de usuario del diseño, se invirtió el tiempo exclusivamente necesario en elaboración de diagramas y diseño de interfaz gráfica, elaborando una aplicación sencilla e intuitiva, pero que cumpla los requerimientos del usuario, desde el punto de vista de las interfaces, se invirtió el tiempo necesario en su diseño, se prestó mucha atención al elaborar formularios, efectos, ubicación de los elementos tal y como el cliente las había solicitado, obteniendo respuesta positiva del cliente, manifestando conformidad con la apariencia visual de la aplicación.

## 2.2. Glosario de Términos

TABLA XXVIII : GLOSARIO DE TÉRMINOS

<b>Término</b>	<b>Definición</b>
Administrador	Usuario que se encuentra encargado de la administración del software.
Administrador farmacia	Usuario que se encuentra a cargo de todo el departamento de Farmacia.
Vendedor	Usuario que solo se encarga de emitir ventas, transferencias, anular y despachar reportes de hospitalización.
Cliente	Persona que ha realizado una compra en farmacia.
Factura	Comprobante de que se ha realizado una compra en esta dependencia.
Receta	Informe que el doctor ha recetado medicamento para que sea facturado en Farmacia
Transferencia	Comprobante que se emite par la entrega de medicamento a otras dependencias ejemplo quirófano, rayos x, etc.
Hospitalización	Comprobante de consumo de medicamento de un paciente que se encuentre hospitalizado.
Proveedor	Persona o Empresa que provee de productos a esta institución.
Canje	Proceso que se realiza en el caso que el producto este al caducar y selo realiza por medio del proveedor.
Kardex	Informe de movimiento de producto ya sea por medio de

	venta, transferencia u hospitalización.
Inventario	Lista de producto que se encuentra disponible en farmacia.
Liquidación	Informe económico de gastos mensuales y venta mensual para dar a conocer la utilidad o ganancia de esta institución.
Cierre de caja diario	Informe de la venta diaria que se hizo.
Producto	Elemento que ingresa a formar parte del inventario para poder ser comercializado.

### 2.3. Tarjetas CRC (Clase - Responsabilidad – Colaborador)

TABLA XXIX : TARJETAS CRC CLIENTE

CLIENTE	
Ingresar cliente	Cliente
Modificar cliente	información_adicional_paciente

TABLA XXX : TARJETAS CRC PRODUCTO

PRODUCTO	
Ingresar producto	Factura_compra

TABLA XXXI : TARJETAS CRC PROVEEDOR

PROVEEDOR	
Ingresar proveedor	proveedor
Modificar proveedor	proveedor

TABLA XXXII : TARJETAS CRC FACTURA DE COMPRA

FACTURA_COMPRA	
Ingresar proveedor	Proveedor
Ingresar producto	Producto
	Ingreso_producto
	Kardex
	Línea_kardex

TABLA XXXIII : TARJETAS CRC FACTURA DE VENTA

FACTURA_VENTA	
Ingresar cliente	Cliente
Ingresar producto	Item_factura
	Producto
	Kardex
	Línea_kardex

TABLA XXXIV : TARJETAS CRC TRANSFERENCIAS

TRANSPASO	
Ingresar producto	Item_traspaso
	Producto
	Kardex
	Línea_kardex

TABLA XXXV : TARJETAS CRC HOSPITALIZACIÓN

ITEM_HOSPITALIZACIÓN	
Despachar medicamento	Producto
	Carde
	Línea_kardex

TABLA XXXVI : TARJETAS CRC FACTURAR RECETAS

FACTURAR RECETAS	
Facturar recetas	Cliente Proforma Ítem_proforma Producto Ingreso_producto Kardex Línea_kardex

TABLA XXXVII : TARJETAS CRC CANJE

CANJE	
Canjear producto	Canje Producto Ingreso_producto Kardex Línea_kardex

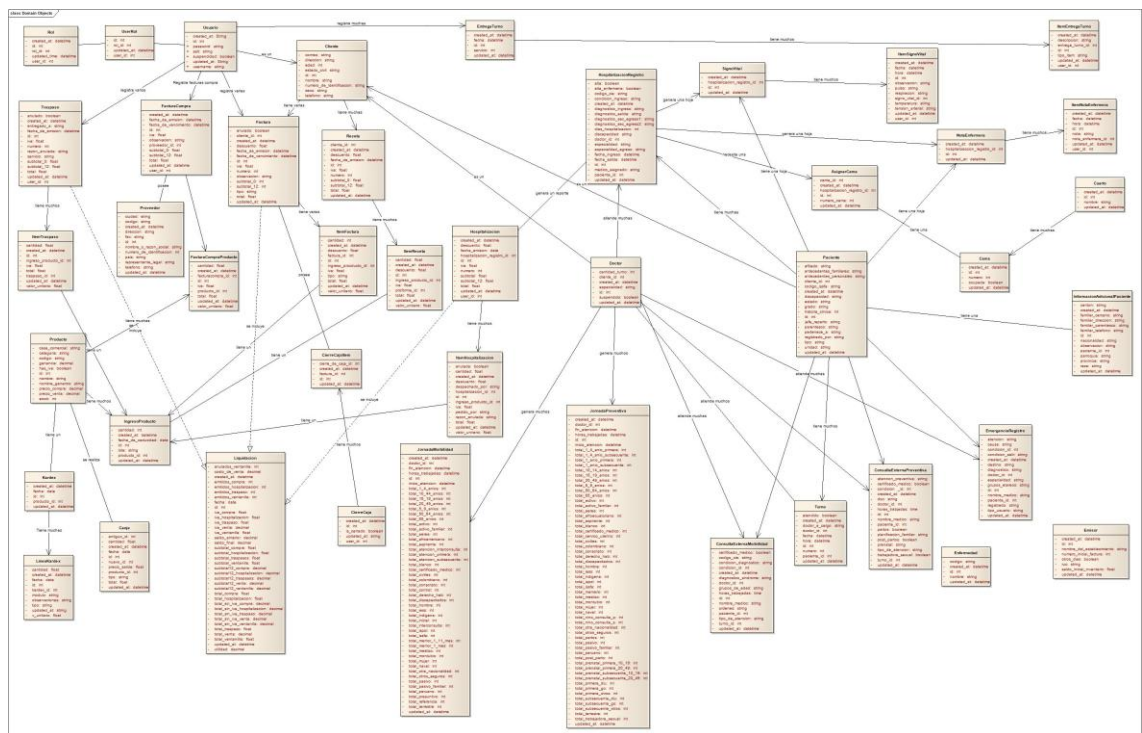
TABLA XXXVIII : TARJETAS CRC CIERRE DE CAJA

CIERRE_CAJA	
Cerrar caja	Cierre_caja Cierre_caja_item



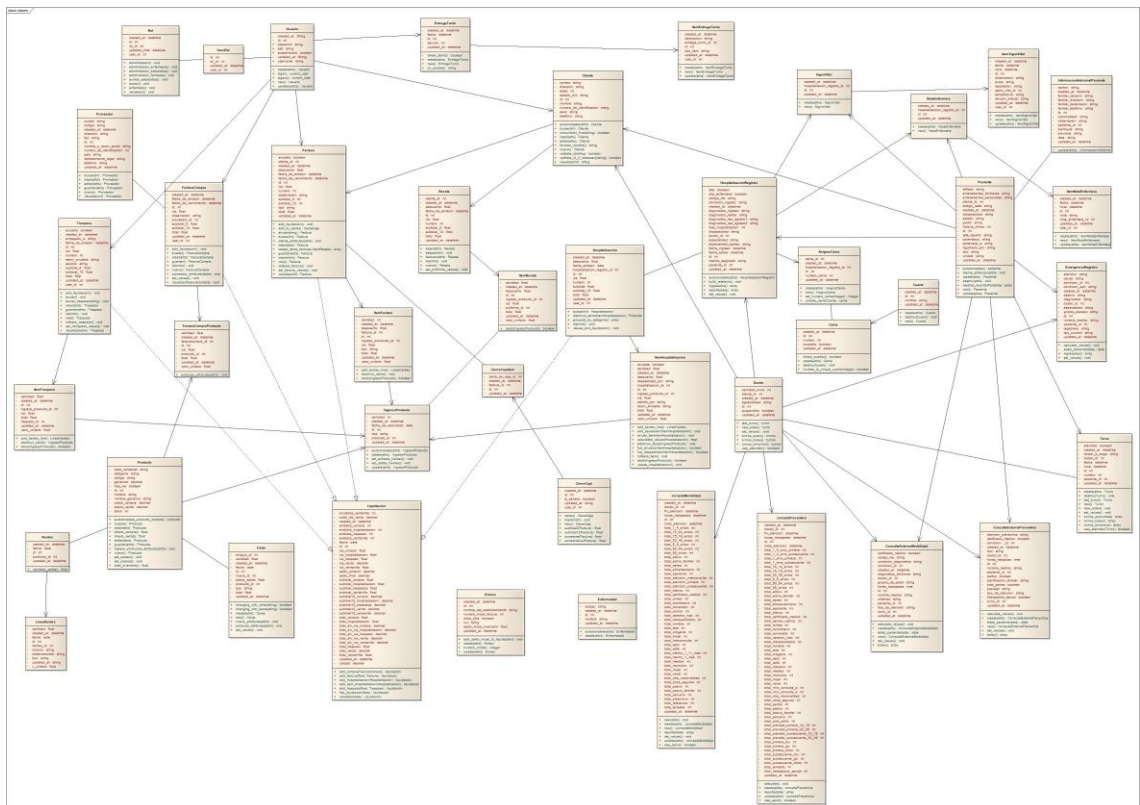
## 2.4. Modelo del Dominio

El modelo de dominio se crea con el fin de representar el vocabulario y los conceptos clave del dominio del problema. El modelo de dominio también identifica las relaciones entre todas las entidades comprendidas en el ámbito del dominio del problema, y comúnmente identifica sus atributos. Un modelo de dominio que encapsula los métodos dentro de las entidades se asocia más bien con modelos orientados a objetos.



## 2.5. Modelo de Clase.

El diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. El diagrama de clases fue utilizado durante el proceso de análisis y diseño del sistema, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la relación entre uno y otro. En este diagrama de clases se pueden distinguir principalmente dos elementos: clases y sus relaciones.



## 2.6. Modelo de la Base de Datos.

El modelo de base de datos es un tipo de modelo de datos que determina la estructura lógica de la base de datos y de manera fundamental determina el modo de almacenar, organizar y manipular los datos que se van a compartir en el sistema, para mejor entendimiento de este tipo de sistemas y modelado.



### 3. FASE 3.- Desarrollo

En esta metodología la codificación se realiza desde el principio, lo cual permite favorablemente el logro de los objetivos, por medio de las entregas frecuentemente al cliente. Los elementos más importantes en cuanto a la codificación son: el cliente siempre debe estar presente, se debe trabajar en parejas y debe haber una propiedad colectiva del código.

#### 3.1. Cliente Siempre Presente

Al tener al cliente presente mientras se desarrolla la aplicación, es un punto algo complejo de asimilar si se consideran los costos que esto representa, por tal motivo se llevó comunicación constante, en la cual el desarrollador debía acudir al hospital, podía llamar vía telefónica al cliente o mediante el internet, en el momento que requieran solucionar cualquier duda en el proceso de implementación. De esta forma se logra una buena comunicación con el cliente.

#### 3.2. Pruebas Unitarias

Estas pruebas se las puede correr en modo desarrollo, las cuales se encuentra en el repositorio de la aplicación. A continuación se presenta en tablas el detalle de las mismas.

TABLA XXXIX : PRUEBA UNITARIA 1

<b>Número de Caso de Prueba:</b> 01
<b>Caso de Prueba:</b> Número de identificación es único
<b>Objetivo de la prueba:</b> Verificar el número de identificación se encuentra registrado en el sistema y que no es posible registrarse nuevamente.
<b>Condiciones:</b> Estar registrado en el sistema.
<b>Datos de Entrada:</b> Datos del usuario específicamente número de identificación.
<b>Salida Esperada:</b> Informar que se encuentra en el sistema.
<b>Salida Obtenida:</b> El usuario existe en el sistema
<b>Evaluación:</b> Prueba satisfactoria

TABLA XL : PRUEBA UNITARIA 2

<b>Número de Caso de Prueba:</b> 02
<b>Caso de Prueba:</b> Caducidad de producto
<b>Objetivo de la prueba:</b> El producto ingresado por medio de una factura de compra tiene que ser mayor a 5 meses de la fecha que se ingresa.
<b>Condiciones:</b> Ser administrador de farmacia
<b>Datos de Entrada:</b> datos del producto
<b>Salida Esperada:</b> informar que el producto no puede ser almacenado
<b>Salida Obtenida:</b> fecha invalida
<b>Evaluación:</b> Prueba satisfactoria

TABLA XLI : PRUEBA UNITARIA 3

<b>Número de Caso de Prueba:</b> 03
<b>Caso de Prueba:</b> cierre de caja por usuario
<b>Objetivo de la prueba:</b> Verificar que el cierre de caja se lo hago por usuarios, en otras palabras cada usuario tiene q visualizar sus ventas nada mas no la de los demás vendedores ni del administrador de farmacia.
<b>Condiciones:</b> Estar registrado en el sistema como administrador de farmacia, o vendedor y haber hecho por lo menos una factura de venta
<b>Datos de Entrada:</b> Factura de venta
<b>Salida Esperada:</b> Informe de cuanto se vendió en transcurso de la jornada
<b>Salida Obtenida:</b> total de cierre de caja
<b>Evaluación:</b> Prueba satisfactoria

TABLA XLII : PRUEBA UNITARIA 4

<b>Número de Caso de Prueba:</b> 04
<b>Caso de Prueba:</b> Factura de compra
<b>Objetivo de la prueba:</b> Verificar que no se emita una factura de compra sin haber registrado el número de la factura
<b>Condiciones:</b> Estar registrado en el sistema como administrador de farmacia, que se haya ingresado un proveedor y por lo menos un producto.
<b>Datos de Entrada:</b> producto, proveedor, numero de factura
<b>Salida Esperada:</b> Informar que no puede ser guardado porque faltan los 3 campos antes mencionados
<b>Salida Obtenida:</b> Faltan campos obligatorios
<b>Evaluación:</b> Prueba satisfactoria

### **3.3. Programación por parejas**

La metodología XP, recalca que toda la producción de código debe ser hecha en parejas, donde se tiene un diseño de mejor calidad y un código más organizado y se soluciona los problemas más fácilmente. En el desarrollo de este proyecto se creó un repositorio llamado “sisfa” donde se muestra: historial de cambios, tareas y bugs del sistema, llevando de manera ordenada el código.

### **3.4. Integración**

En este punto se hace mención que se deben hacer integraciones cada pocas horas o en lo posible no tardar más de un día entre una y otra integración, entre más se tarda en encontrar un problema, resultará más costoso resolverlo, e integrar frecuentemente evita problemas como el trabajar sobre una clase obsoleta. En cuanto a esta parte no existió problema en realizarlo, debido a que se lleva un control de versiones y una vez aprobadas se realizaba la integración.

## **4. Fase 4.- Pruebas**

### **4.1. Implantación**

Al haber finalizado el desarrollo de la aplicación se la implantó, en un servidor de la Institución que sirvió como pruebas locales, y en un servidor web para su implantación final, verificando su funcionamiento y realizando las respectivas explicaciones al Ing. Juan Pablo Sánchez, quien será el encargado del manejo de el mismo. De esta manera se llevó a cabo con el cumplimiento del objetivo principal del proyecto de fin de carrera.

### **4.2. Pruebas de Aceptación**

Las pruebas de aceptación, son diseñadas en base a las historias de usuario, es una prueba formal conducida para determinar si un sistema satisface los criterios de aceptación y permite al cliente determinar si acepta el sistema. Para lo cual se elaboró las siguientes tablas.

TABLA XLIII : CASO DE PRUEBA 1: ADMINISTRACIÓN USUARIOS

<b>Número de Caso de Prueba: 01</b>	<b>Número de Historia de Usuario: 01</b>
<b>Caso de Prueba:</b> Administración usuarios	
<b>Nombre de Caso de Prueba:</b> Ingresar Usuarios Correctamente	
<b>Descripción:</b> Verificará que el administrador puede realizar ingresos de nuevos usuarios.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado correctamente los datos requeridos para ingresar.	
<b>Entrada:</b> 1. Seleccionar el menú Usuarios. 2. Seleccionar el Nuevo Usuario. 3. Llenar correctamente todos campos solicitados. 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al nuevo usuario.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso se ejecutó correctamente.	

TABLA XLIV : CASO DE PRUEBA 2: ADMINISTRACIÓN USUARIOS

<b>Número de Caso de Prueba: 02</b>	<b>Número de Historia de Usuario: 01</b>
<b>Caso de Prueba:</b> Administración usuarios	
<b>Nombre de Caso de Prueba:</b> Ingresar Usuarios Incorrectamente	
<b>Descripción:</b> Verificará que si los datos que ingresa el administrador, no son correctos o no son llenados no se realiza el ingreso del mismo.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado incorrectamente los datos o tiene campos en blanco.	
<b>Entrada:</b> 1. Seleccionar el menú Usuarios. 2. Seleccionar el Nuevo Usuario. 3. Llenar incorrectamente los campos solicitados o dejarlos en blanco. 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> No se guarda la información correspondiente al nuevo usuario. Se presenta mensajes indicando que campo no ha sido llenado o si el tipo de dato es incorrecto.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso no se ejecutó.	

TABLA XLV : CASO DE PRUEBA 3: ADMINISTRACIÓN USUARIOS

<b>Número de Caso de Prueba: 03</b>	<b>Número de Historia de Usuario: 01</b>
<b>Caso de Prueba:</b> Administración usuarios	
<b>Nombre de Caso de Prueba:</b> Suspender usuario	
<b>Descripción:</b> Verificará que el administrador puede suspender usuarios.	
<b>Condiciones de Ejecución:</b> El usuario ha sido registrado previamente.	
<b>Entrada:</b> 1. Seleccionar el menú Usuarios. 2. Buscar usuario. 3. Seleccionar icono suspender. 4. Seleccionar el botón OK.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al usuario.	
<b>Evaluación:</b> El usuario se suspendió correctamente.	

TABLA XLVI : CASO DE PRUEBA 4: ADMINISTRACIÓN USUARIOS

<b>Número de Caso de Prueba: 04</b>	<b>Número de Historia de Usuario: 01</b>
<b>Caso de Prueba:</b> Administración usuarios	
<b>Nombre de Caso de Prueba:</b> Activar usuario	
<b>Descripción:</b> Verificará que el administrador puede activar usuarios.	
<b>Condiciones de Ejecución:</b> El usuario ha sido registrado previamente.	
<b>Entrada:</b> 1. Seleccionar el menú Usuarios. 2. Buscar usuario. 3. Seleccionar icono activar. 4. Seleccionar el botón OK.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al usuario.	
<b>Evaluación:</b> El usuario se activó correctamente.	



TABLA XLVII : CASO DE PRUEBA 5: ADMINISTRACIÓN USUARIOS

<b>Número de Caso de Prueba: 05</b>	<b>Número de Historia de Usuario: 01</b>
<b>Caso de Prueba:</b> Administración usuarios	
<b>Nombre de Caso de Prueba:</b> Nueva clave	
<b>Descripción:</b> Verificará que el administrador puede dar nueva clave a usuarios.	
<b>Condiciones de Ejecución:</b> El usuario ha sido registrado previamente.	
<b>Entrada:</b> 1. Seleccionar el menú Usuarios. 2. Buscar usuario. 3. Seleccionar icono nueva clave. 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al usuario.	
<b>Evaluación:</b> Nueva clave ingresada correctamente.	

TABLA XLVIII : CASO DE PRUEBA 6: ADMINISTRACIÓN CLIENTE

<b>Número de Caso de Prueba: 06</b>	<b>Número de Historia de Usuario: 02</b>
<b>Caso de Prueba:</b> Administración de cliente	
<b>Nombre de Caso de Prueba:</b> Ingresar cliente	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor puedan realizar aperturas de nuevos clientes.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado correctamente los datos requeridos.	
<b>Entrada:</b> 1. Seleccionar el menú Clientes. 2. Seleccionar el icono (+) 3. Llenar correctamente todos campos solicitados. 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al nuevo cliente.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso se ejecutó correctamente.	

TABLA XLIX : CASO DE PRUEBA 7: ADMINISTRACIÓN CLIENTE

<b>Número de Caso de Prueba: 07</b>	<b>Número de Historia de Usuario: 02</b>
<b>Caso de Prueba:</b> Administración de cliente	
<b>Nombre de Caso de Prueba:</b> Ingresar cliente incorrectamente	
<b>Descripción:</b> Verificará que si el administrador de farmacia o vendedor ingresan mal los datos no se registran una historia clínica.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado incorrectamente los datos o tiene campos obligatorios en blanco.	
<b>Entrada:</b> 1. Seleccionar el menú clientes. 2. Seleccionar el icono (+) 3. Llenar incorrectamente los campos o dejar campos en blanco. 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> No se guarda la información correspondiente al paciente. Se presenta mensajes indicando que campo no ha sido llenado o si el tipo de dato es incorrecto.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso no se ejecutó.	

TABLA L : CASO DE PRUEBA 8: ADMINISTRACIÓN CLIENTES

<b>Número de Caso de Prueba: 08</b>	<b>Número de Historia de Usuario: 02</b>
<b>Caso de Prueba:</b> Administración de clientes	
<b>Nombre de Caso de Prueba:</b> Modificar clientes	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor puedan realizar modificaciones de un cliente.	
<b>Condiciones de Ejecución:</b> El cliente ha sido ingresado previamente.	
<b>Entrada:</b> 1. Seleccionar el menú Clientes. 2. Buscar cliente. 3. Seleccionar botón editar. 4. Seleccionar el campo que desea. 5. Modificar el campo seleccionado. 6. Click en el botón guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al cliente.	
<b>Evaluación:</b> Se modificó los datos correctamente.	

TABLA LI : CASO DE PRUEBA 9: ADMINISTARCIÓN DE PROVEEDORES

<b>Número de Caso de Prueba: 9</b>	<b>Número de Historia de Usuario: 03</b>
<b>Caso de Prueba:</b> Administración de proveedores	
<b>Nombre de Caso de Prueba:</b> Crear proveedores	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor puedan realizar aperturas de nuevos proveedores.	
<b>Condiciones de Ejecución:</b> El usuario ha ingresado correctamente los datos requeridos.	
<b>Entrada:</b> 1. Seleccionar el menú Proveedores 2. Seleccionar el icono (+) 3. Llenar correctamente todos campos solicitados. 4. Dar Click en el botón guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al proveedor	
<b>Evaluación:</b> Datos guardados satisfactoriamente.	

TABLA LII : CASO DE PRUEBA 10: ADMINISTRACIÓN DE PROVEEDORES

<b>Número de Caso de Prueba: 10</b>	<b>Número de Historia de Usuario: 03</b>
<b>Caso de Prueba:</b> Administración de proveedores	
<b>Nombre de Caso de Prueba:</b> Ingresar proveedor incorrectamente	
<b>Descripción:</b> Verificará que si el administrador de farmacia o vendedor ingresan mal los datos y no se guarde la información.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado incorrectamente los datos o tiene campos obligatorios en blanco.	
<b>Entrada:</b> 1. Seleccionar el menú proveedores. 2. Seleccionar el icono (+) 3. Llenar incorrectamente los campos o dejar campos en blanco. 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> No se guarda la información correspondiente al paciente. Se presenta mensajes indicando que campo no ha sido llenado o si el tipo de dato es incorrecto.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso no se ejecutó.	

TABLA LIII : CASO DE PRUEBA 11: ADMINISTRAR PROVEEDORES

<b>Número de Caso de Prueba: 11</b>	<b>Número de Historia de Usuario: 03</b>
<b>Caso de Prueba:</b> Administración de proveedores	
<b>Nombre de Caso de Prueba:</b> Modificar proveedor	
<b>Descripción:</b> Verificará que el administrador de farmacia pueda realizar modificaciones de un proveedor	
<b>Condiciones de Ejecución:</b> El proveedor ha sido ingresado previamente.	
<b>Entrada:</b> 1. Seleccionar el menú proveedores. 2. Buscar proveedor 3. Seleccionar botón editar. 4. Seleccionar el campo que desea. 5. Modificar el campo seleccionado. 6. Click en el botón guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al proveedor.	
<b>Evaluación:</b> Se modificó los datos correctamente	

TABLA LIV : CASO DE PRUEBA 12: ADMINISTRAR PRODUCTO

<b>Número de Caso de Prueba: 12</b>	<b>Número de Historia de Usuario: 04</b>
<b>Caso de Prueba:</b> Crear producto	
<b>Nombre de Caso de Prueba:</b> Ingresar producto correctamente	
<b>Descripción:</b> Verificará que el administrador de farmacia pueda realizar ingreso de nuevos productos.	
<b>Condiciones de Ejecución:</b> El usuario ha ingresado correctamente los datos requeridos para el producto	
<b>Entrada:</b> 1. Seleccionar el menú Factura de compra 2. Llenar los campos del proveedor. 3. Llenar correctamente todos campos solicitados de producto. 4. Dar Click en el botón guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al producto	
<b>Evaluación:</b> Datos guardados satisfactoriamente.	

TABLA LV : CASO DE PRUEBA 13: ADMINISTRAR PRODUCTO

<b>Número de Caso de Prueba: 13</b>	<b>Número de Historia de Usuario: 04</b>
<b>Caso de Prueba:</b> Administración de proveedores	
<b>Nombre de Caso de Prueba:</b> Ingresar producto incorrectamente	
<b>Descripción:</b> Verificará que si el administrador de farmacia ingresa mal los datos y no se guarde la información.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado incorrectamente los datos o tiene campos obligatorios en blanco.	
<b>Entrada:</b> 1. Seleccionar el menú Factura de compra 2. Llenar los campos del proveedor incorrectamente. 3. Llenar correctamente todos campos solicitados de producto incorrectamente. 4. Dar Click en el botón guardar.	
<b>Resultado Esperado:</b> No se guarda la información correspondiente al producto. Se presenta mensajes indicando que campo no ha sido llenado o si el tipo de dato es incorrecto.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso no se ejecutó.	

TABLA LVI : CASO DE PRUEBA 14: ADMINISTRAR FACTURA DE VENTA

<b>Número de Caso de Prueba: 14</b>	<b>Número de Historia de Usuario: 05</b>
<b>Caso de Prueba:</b> Administrar Factura De Venta	
<b>Nombre de Caso de Prueba:</b> Ingresar factura de venta	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor puedan emitir una factura a un cliente.	
<b>Condiciones de Ejecución:</b> El usuario ha ingresado correctamente los datos requeridos.	
<b>Entrada:</b> 1. Seleccionar el menú Facturas de venta. 2. Llenar los campos del cliente. 3. Llenar correctamente todos campos solicitados del producto. 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> Se actualizó la información.	
<b>Evaluación:</b> Factura guardada satisfactoriamente.	

TABLA LVII : CASO DE PRUEBA 15: ADMINISTRAR FACTURA DE VENTA

<b>Número de Caso de Prueba: 15</b>	<b>Número de Historia de Usuario: 05</b>
<b>Caso de Prueba:</b> Administrar Factura De Venta	
<b>Nombre de Caso de Prueba:</b> Ingresar factura de venta incorrectamente	
<b>Descripción:</b> Verificará que si el administrador de farmacia y el vendedor ingresen mal los datos y no se guarde la información.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado incorrectamente los datos o tiene campos obligatorios en blanco.	
<b>Entrada:</b> 1. Seleccionar el menú Facturas de venta. 2. Llenar incorrectamente datos del cliente. 3. Llenar incorrectamente todos campos solicitados del producto. 4. Seleccionar el botón guardar	
<b>Resultado Esperado:</b> No se guarda la información correspondiente al producto. Se presenta mensajes indicando que campo no ha sido llenado o si el tipo de dato es incorrecto.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso no se ejecutó.	

TABLA LVIII : CASO DE PRUEBA 16: ADMINISTRACIÓN DE FACTURAS DE VENTA

<b>Número de Caso de Prueba: 16</b>	<b>Número de Historia de Usuario: 05</b>
<b>Caso de Prueba:</b> Administración de facturas de venta	
<b>Nombre de Caso de Prueba:</b> Anular facturas	
<b>Descripción:</b> Verificará que el administrador de farmacia y el vendedor puedan realizar anulaciones de una factura de venta	
<b>Condiciones de Ejecución:</b> La factura ha sido ingresada previamente.	
<b>Entrada:</b> 1. Seleccionar el menú Facturas de venta. 2. Seleccionar botón lista facturas. 3. Seleccionar el botón anular factura 3. Llenar campo de razón por la que se anula 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente a la factura y al producto	
<b>Evaluación:</b> Se anuló la factura satisfactoriamente	

TABLA LIX : CASO DE PRUEBA 17: ADMINISTRACIÓN DE TRANSFERENCIAS

<b>Número de Caso de Prueba: 17</b>	<b>Número de Historia de Usuario: 06</b>
<b>Caso de Prueba:</b> Administracion De Transferencias	
<b>Nombre de Caso de Prueba:</b> Ingresar transferencia	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor puedan emitir una transferencia a una dependencia del hospital.	
<b>Condiciones de Ejecución:</b> El usuario ha ingresado correctamente los datos requeridos.	
<b>Entrada:</b> 1. Seleccionar el menú Transferencias. 2. Llenar los campos de la dependencia 3. Llenar los campos del producto a transferir 4. Seleccionar el botón guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al producto	
<b>Evaluación:</b> La transferencia se guardó correctamente.	

TABLA LX : CASO DE PRUEBA 18: ADMINISTRACIÓN DE TRANSFERENCIAS

<b>Número de Caso de Prueba: 18</b>	<b>Número de Historia de Usuario: 06</b>
<b>Caso de Prueba:</b> Administración de transferencias	
<b>Nombre de Caso de Prueba:</b> Ingresar transferencia incorrectamente	
<b>Descripción:</b> Verificará que si el administrador de farmacia y el vendedor ingresen mal los datos y no se guarde la información.	
<b>Condiciones de Ejecución:</b> El administrador ha ingresado incorrectamente los datos o tiene campos obligatorios en blanco.	
<b>Entrada:</b> 1. Seleccionar el menú Transferencias. 2. Llenar los campos de la dependencia incorrectamente 3. Llenar los campos del producto a transferir incorrectamente 4. Seleccionar el botón guardar.	
<b>Resultado Esperado:</b> No se guarda la información correspondiente al producto. Se presenta mensajes indicando que campo no ha sido llenado o si el tipo de dato es incorrecto.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso no se ejecutó.	

TABLA LXI : CASO DE PRUEBA 19: ADMINISTRACIÓN DE TRANSFERENCIAS

<b>Número de Caso de Prueba: 19</b>	<b>Número de Historia de Usuario: 06</b>
<b>Caso de Prueba:</b> Administración transferencia	
<b>Nombre de Caso de Prueba:</b> Modificar Transferencia	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor pueda realizar modificaciones de una transferencia	
<b>Condiciones de Ejecución:</b> La transferencia ha sido ingresada previamente.	
<b>Entrada:</b> 1. Seleccionar el menú transferencias. 2. Buscar transferencia 3. Seleccionar botón editar. 4. Seleccionar el campo que desea. 5. Modificar el campo seleccionado. 6. Click en el botón guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente a la transferencia.	
<b>Evaluación:</b> Se modificó los datos correctamente	

TABLA LXII : CASO DE PRUEBA 20: ADMINISTRACIÓN DE TRANSFERENCIAS

<b>Número de Caso de Prueba: 20</b>	<b>Número de Historia de Usuario: 06</b>
<b>Caso de Prueba:</b> Administración de transferencias	
<b>Nombre de Caso de Prueba:</b> Anular transferencia	
<b>Descripción:</b> Verificará que el administrador de farmacia y el vendedor puedan realizar anulaciones de una transferencia	
<b>Condiciones de Ejecución:</b> Se actualiza la información correspondiente a la transferencia y al producto	
<b>Entrada:</b> 1. Seleccionar el menú transferencias 2. Seleccionar botón lista transferencias. 3. Seleccionar el botón anular transferencias 3. Llenar campo de razón por la que se anula 4. Seleccionar el botón Guardar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente a la transferencia y al producto	
<b>Evaluación:</b> Se anuló la factura satisfactoriamente	



TABLA LXIII : CASO DE PRUEBA 21: ADMINISTRACIÓN HOSPITALIZACIÓN

<b>Número de Caso de Prueba: 21</b>	<b>Número de Historia de Usuario: 07</b>
<b>Caso de Prueba:</b> Administración Hospitalización	
<b>Nombre de Caso de Prueba:</b> Despachar hospitalización	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor puedan despachar una hospitalización	
<b>Condiciones de Ejecución:</b> El usuario ha ingresado correctamente los datos requeridos.	
<b>Entrada:</b> 1. Seleccionar el menú Hospitalización. 2. Buscar el paciente 3. Despachar producto 4. Click en despachar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente a la hospitalización	
<b>Evaluación:</b> Se despachó correctamente	

TABLA LXIV : CASO DE PRUEBA 22 ADMINISTRACIÓN HOSPITALIZACIÓN

<b>Número de Caso de Prueba: 22</b>	<b>Número de Historia de Usuario: 07</b>
<b>Caso de Prueba:</b> Administración Hospitalización	
<b>Nombre de Caso de Prueba:</b> Modificar despacho de hospitalización	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor pueda realizar modificaciones de un despacho de hospitalización.	
<b>Condiciones de Ejecución:</b> El despacho de hospitalización haya sido ingresado previamente.	
<b>Entrada:</b> 1. Seleccionar el menú Hospitalización. 2. Buscar el paciente 3.click en Modificar el despacho de producto 4. Clic en despachar.	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente al despacho	
<b>Evaluación:</b> Se modificó los datos correctamente	

TABLA LXV : CASO DE PRUEBA 23 ADMINISTRACIÓN DE HOSPITALIZACIÓN

<b>Número de Caso de Prueba: 23</b>	<b>Número de Historia de Usuario: 07</b>
<b>Caso de Prueba:</b> Administración de hospitalización	
<b>Nombre de Caso de Prueba:</b> Anular Hospitalización	
<b>Descripción:</b> Verificará que el administrador de farmacia y el vendedor puedan realizar anulaciones de una HOSPITALIZACION	
<b>Condiciones de Ejecución:</b> Se actualiza la información correspondiente a la hospitalización y al producto	
<b>Entrada:</b> 1. Seleccionar el menú Hospitalización. 2. Buscar el paciente 3. Click en el botón Anular el despacho de producto 4. Llenar motivo de anulación 5. Guardar anulación	
<b>Resultado Esperado:</b> Se actualiza la información correspondiente a la hospitalización y al producto	
<b>Evaluación:</b> Se anuló la hospitalización satisfactoriamente	

TABLA LXVI : CASO DE PRUEBA 24: FACTURAR RECETAS

<b>Número de Caso de Prueba: 24</b>	<b>Número de Historia de Usuario: 08</b>
<b>Caso de Prueba:</b> Facturar recetas	
<b>Nombre de Caso de Prueba:</b> Despachar receta	
<b>Descripción:</b> Verificará que el administrador de farmacia y vendedor puedan emitir una factura a un cliente.	
<b>Condiciones de Ejecución:</b> El doctor ha ingresado correctamente los datos requeridos.	
<b>Entrada:</b> 1. Seleccionar el menú Facturar recetas. 2. Buscar el paciente. 3. Seleccionar el botón mostrar. 4. Ver el producto y facturar.	
<b>Resultado Esperado:</b> Se actualizo la información.	
<b>Evaluación:</b> Factura guardada satisfactoriamente.	

TABLA LXVII : CASO DE PRUEBA 25: ADMINISTRAR CANJE

<b>Número de Caso de Prueba: 25</b>	<b>Número de Historia de Usuario: 09</b>
<b>Caso de Prueba:</b> Administrar canje.	
<b>Nombre de Caso de Prueba:</b> ingresar canje	
<b>Descripción:</b> Verificará que el administrador de farmacia pueda realizar el canje.	
<b>Condiciones de Ejecución:</b> Debe de existir al menos un producto caducado para el canje de producto.	
<b>Entrada:</b> 1. Seleccionar el menú Caducados. 2. Seleccionar canje de producto. 3. Ingresar datos del producto nuevo. 4. Verificar el monto se igual al producto a canjear. 5. click canje.	
<b>Resultado Esperado:</b> Se actualizo la información correctamente	
<b>Evaluación:</b> Canje realizado con éxito	

TABLA LXVIII : CASO DE PRUEBA 26: ADMINISTRAR CANJE

<b>Número de Caso de Prueba: 26</b>	<b>Número de Historia de Usuario: 09</b>
<b>Caso de Prueba:</b> Administrar canje	
<b>Nombre de Caso de Prueba:</b> Ingresar canje incorrectamente	
<b>Descripción:</b> Verificará que si el administrador de farmacia ingrese mal los datos y no se guarde la información.	
<b>Condiciones de Ejecución:</b> Debe de existir al menos un producto caducado.	
<b>Entrada:</b> 1. Seleccionar el menú Caducados. 2. Seleccionar canje de producto. 3. Ingresar datos del producto nuevo incorrectamente. 4. Click canje.	
<b>Resultado Esperado:</b> No se guarda la información correspondiente al producto. Se presenta mensajes indicando que campo no ha sido llenado o si el tipo de dato es incorrecto.	
<b>Evaluación:</b> La evaluación fue aprobada, el ingreso no se ejecutó.	

TABLA LXIX : CASO DE PRUEBA 27: CIERRE DE CAJA

<b>Número de Caso de Prueba: 27</b>	<b>Número de Historia de Usuario: 10</b>
<b>Caso de Prueba:</b> Cierre de caja	
<b>Nombre de Caso de Prueba:</b> Realizar cierre de caja	
<b>Descripción:</b> Verificará que el administrador de farmacia y el vendedor puedan realizar un cierre de caja	
<b>Condiciones de Ejecución:</b> Que por lo menos haya factura una vez en su turno de trabajo.	
<b>Entrada:</b> 1. Seleccionar el menú Cierre de caja. 2. Seleccionar la opción deseo cerrar caja. 3. Click en cerrar caja	
<b>Resultado Esperado:</b> Se muestra la cantidad de venta hechas durante el periodo de trabajo con su respectiva información.	
<b>Evaluación:</b> El cierre de caja se realizó con éxito	

TABLA LXX : CASO DE PRUEBA 28: REPORTES FINANCIEROS MENSUALES

<b>Número de Caso de Prueba: 28</b>	<b>Número de Historia de Usuario: 11</b>
<b>Caso de Prueba:</b> Reportes financieros mensuales	
<b>Nombre de Caso de Prueba:</b> Realizar reportes de venta, transferencia, hospitalización y compras	
<b>Descripción:</b> Verificará que el administrador de farmacia pueda realizar un reporte	
<b>Condiciones de Ejecución:</b> Debe de haber facturas, transferencias u hospitalizaciones realizadas en cualquier periodo dependiendo la consulta que realiza el administrador	
<b>Entrada:</b> 1. Seleccionar el menú reportes. 2. Seleccionar pestaña y escoger la opción correspondiente. 3. Click en descargar PDF. 3. Seleccionar imprimir.	
<b>Resultado Esperado:</b> Indica un informe detallado de la opción escogida	
<b>Evaluación:</b> Documento impreso satisfactoriamente	

TABLA LXXI : CASO DE PRUEBA 29: REPORTES FINANCIEROS MENSUALES

<b>Número de Caso de Prueba: 29</b>	<b>Número de Historia de Usuario: 12</b>
<b>Caso de Prueba:</b> Reportes financieros mensuales	
<b>Nombre de Caso de Prueba:</b> Realizar reportes de cierre de caja mensual y por días	
<b>Descripción:</b> Verificará que el administrador de farmacia pueda realizar un reporte	
<b>Condiciones de Ejecución:</b> Debe de haber facturas realizadas en cualquier periodo dependiendo la consulta que realiza el administrador	
<b>Entrada:</b> 1. Seleccionar el menú reportes. 2. Seleccionar la opción de mes o día y escoger la opción correspondiente. 3. Dar click en el botón ir. 4. Click en descargar PDF. 4. Seleccionar imprimir	
<b>Resultado Esperado:</b> Indica un informe detallado de la opción escogida de un cierre de caja determinado	
<b>Evaluación:</b> Documento impreso satisfactoriamente	

TABLA LXXII : CASO DE PRUEBA 30: REPORTES FINANCIEROS MENSUALES

<b>Número de Caso de Prueba: 30</b>	<b>Número de Historia de Usuario: 13</b>
<b>Caso de Prueba:</b> Reportes financieros mensuales	
<b>Nombre de Caso de Prueba:</b> Realizar reportes de liquidación mensual	
<b>Descripción:</b> Verificará que el administrador de farmacia pueda realizar un reporte	
<b>Condiciones de Ejecución:</b> Debe de haber facturas, transferencias, hospitalizaciones y compras realizadas en cualquier periodo dependiendo la consulta que realiza el administrador	
<b>Entrada:</b> 1. Seleccionar el menú reportes. 2. Seleccionar la opción del mes a consultar. 3. Click en el botón ir. 4. Click en descargar PDF. 3. Seleccionar imprimir.	
<b>Resultado Esperado:</b> Indica un informe detallado de compras, ventas, transferencias, y hospitalizaciones y la utilidad o ganancia del hospital en el mes.	
<b>Evaluación:</b> Documento impreso satisfactoriamente	

### **4.3. Plan de Validación**

#### **4.3.1. Propósito**

En el desarrollo del software se ve implicado la realización de un conjunto de actividades las cuales tienen como finalidad comprobar el correcto funcionamiento del software realizado, además del cumplimiento de los requerimientos encontrados durante la etapa de planificación. Las pruebas o también conocido como testing son un proceso de ejecución del sistema que buscan descubrir la mayor cantidad de errores en el código fuente.

Al realizar las pruebas estas garantizan que los usuarios reciban una herramienta confiable. Para realizar las pruebas del software, se usó el servidor de la Institución.

#### **4.3.2. Introducción**

Las pruebas de software permiten identificar, verificar y revelar la calidad de un producto de software. Estas nos indican los posibles fallos de implementación, calidad o usabilidad. Las pruebas se integran dentro de diferentes fases del ciclo de vida de software, así se ejecuta un programa y mediante técnicas experimentales se trata de descubrir que errores tiene.

##### **4.3.2.1. Objetivos**

El plan de pruebas para el sistema del Hospital Básico 7 B.I Loja deben procurar los siguientes objetivos:

- Detallar las actividades requeridas para la preparación y conducción de las pruebas al sistema.
- Definir el origen de la información usada para la preparación de las pruebas
- Definir las herramientas para pruebas y el ambiente necesario para su ejecución

##### **4.3.2.2. Antecedentes**

Las especificaciones del sistema del proyecto en el documento se encuentran en resultados, fase 1 del presente informe.

#### 4.3.2.3. Alcances

Este plan de pruebas detalla los procedimientos de los módulos del sistema.

#### 4.3.2.4. Referencias

Los siguientes documentos fueron usados como fuente de información para la preparación de las pruebas.

Especificación de requerimientos de software (Resultados, Fase 1)

Historias de usuario (Resultados, Fase 1, punto 3.4).

Manual de usuario (se encuentra adjunto con este informe)

Manual del programador (se encuentra adjunto con este informe)

### 1. Elementos software a probar

Los módulos que serán probados en este plan de pruebas son los siguientes

TABLA LXXIII : MÓDULOS A PRUEBA

Categoría	Funcionalidad
Administrador	Acceder al sistema
	Administrar usuarios
Administrador Farmacia	Administrar producto
	Administrar clientes
	Administrar proveedores
	Realizar transferencias
	Realizar hospitalizaciones
	Realizar facturas de venta
	Realizar facturas de compra
	Facturar recetas
	Realizar cierres de caja
	Realizar reportes mensuales
Vendedor	Administrar clientes
	Realizar facturas de venta
	Realizar hospitalizaciones
	Realizar transferencias
	Facturar recetas
	Realizar cierres de caja
	Ver producto
	Ver caducados

#### **4.3.4. Características a probar**

Las características que serán probadas son las siguientes:

1. Accesos al sistema.
2. Administración de clientes.
3. Administración de producto
4. Administración de proveedores.
5. Realizar transferencias
6. Realizar hospitalizaciones.
7. Realizar facturas de venta.
8. Realizar facturas de compra.
9. Realizar cierres de caja
10. Realizar reportes mensuales

#### **4.3.5. Enfoque general de la prueba**

##### **4.3.5.1. Aspectos generales**

###### **4.3.5.1.1. En lo relativo a la documentación**

Se usa como referencia los documentos indicados en el apartado 3.4. (Historias de Usuario).

###### **4.3.5.1.2. En lo relativo a los encargados del desarrollo**

El desarrollo de los procedimientos de los módulos del sistema, fueron de la mano con el diseño de los casos de prueba (apartado 4.2., de resultados) y las historias de usuario.

##### **4.3.5.2. Enfoques**

###### **4.3.5.2.1. Enfoque para probar los accesos al sistema**

Realizar múltiples accesos al sistema en forma concurrente con los diferentes tipos de usuarios.



#### **4.3.5.2.2. Enfoque para probar administración de clientes**

Realizar la creación de un cliente, y modificar sus datos.

#### **4.3.5.2.3. Enfoque para probar administración de producto**

Realizar la creación de un producto.

#### **4.3.5.2.4. Enfoque para probar administracion de proveedores.**

Realizar la creación de un proveedor, y modificar sus datos.

#### **4.3.5.2.5. Enfoque para probar, realizar transferencias**

Crear, modificar y anular transferencias.

#### **4.3.5.2.6. Enfoque para probar, realizar hospitalizaciones**

Crear, modificar y anular hospitalizaciones

#### **4.3.5.2.7. Enfoque para probar facturas de venta**

Crear, anular facturas de venta por parte del administrador y vendedor

#### **4.3.5.2.8. Enfoque para probar facturas de compra**

Crear o ingresar el producto comprado y automáticamente se carga al inventario con su respectivo kardex.

#### **4.3.5.2.9. Enfoque para probar recetas.**

Despacho de la receta emitida por el doctor.

#### **4.3.5.2.10. Enfoque para probar cierres de caja**

Emitir facturas durante el día y luego se procede a cerrar caja y saber el efectivo hecho durante el día.

#### **4.3.5.2.11. Enfoque para probar reportes mensuales**

Emitir un reporte de las vetas, hospitalizaciones, transferencias o compras que se realizó durante el mes, adema un reporte detallado de la liquidación mensual con su respectiva utilidad.

#### **4.3.5.3. Amplitud de la prueba**

Cada módulo se refiere una serie de procesos y tendrá asociado las pruebas de aceptación y funcionalidad.

#### **4.3.6. Criterios de paso/fallo para cada elemento**

Los criterios que determinan el éxito o fallo de una prueba se describirán en las especificaciones de casos de pruebas correspondientes.

#### **4.3.7. Criterios de suspensión y requisitos de reanudación**

Una prueba deberá ser suspendida si el servidor no está funcionando apropiadamente.

#### **4.3.8. Productos entregables**

Los siguientes documentos son los generados:

1. Pruebas funcionales
2. Pruebas de aceptación
3. Pruebas de usabilidad
4. Pruebas de Carga y Estrés

### **4.3.9. Necesidades de entorno**

#### **4.3.9.1. Hardware**

Se requiere un servidor y un computador por usuario.

##### **4.3.9.1.1. Características del servidor**

Computadora CORE I5, 1.0GHz, 6 GB RAM 1 TB Disco Duro.

##### **4.3.9.1.2. Características del computador**

Computadora core i3, 1.20GHz, 4 GB RAM 500 GB Disco Duro.

#### **4.3.9.2. Software**

Sistema operativo Ubuntu Server (u otra versión de Linux), también se puede utilizar el sistema operativo Microsoft Windows XP, W7, server 2008.

Servidor web: Apache 2.2.17 o superior.

Gestor de base de datos MySQL 5.0 o superior

Requerimientos especiales Intérprete de Ruby 2.2. O superior

Sistema operativo Windows, Linux, Mac

Navegador Web (Mozilla Firefox, Google Chrome, Internet Explorer, Opera, etc).

##### **4.3.9.3. Documentos**

El documento necesario es el Manual del Usuario.

### **4.3.10. Responsabilidades en la organización y realización de pruebas**

#### **4.3.10.1. Responsable del proyecto**

El responsable del proyecto seleccionó los módulos a ser probados, luego se realizó la administración y soporte técnico de las pruebas en general.

#### **4.3.10.2. Grupo de usuarios del sistema del Hospital Básico 7 B.I Loja.**

Las pruebas fueron aplicadas al personal del Hospital Básico 7 B.I Loja, 1 encuesta al futuro administrador del sistema y 4 encuestas a posibles usuarios del sistema, siendo un total de 5 encuestas (Ver Anexo 1).

Las pruebas fueron dirigidas con la ayuda del Ing. Juan Pablo Sánchez, encargado del departamento de Sistemas, en primera instancia se realizó algunas explicaciones y una capacitación final del funcionamiento de cada módulo del sistema, indicando cada perfil de usuario y sus respectivas funciones dentro del sistema.

Luego se realizó encuestas para cada perfil de usuario para medir la funcionalidad del sistema. A continuación se detallan cada uno de los procesos evaluados en cada parámetro que permitieron en su totalidad evaluar el sistema.

#### **4.3.11. Pruebas de Funcionalidad:**

Este tipo de pruebas permitió garantizar que la funcionalidad del sistema se dé correctamente, cumpliendo de esta manera con los requerimientos solicitados por el usuario (ver Anexo 2). Los parámetros a evaluar:

Hubo algún problema al realizar alguna de las siguientes actividades:

1. Crear usuario administrador de farmacia
2. Crear usuario vendedor
3. Suspende usuarios.
4. Administrar clientes.
5. Administrar producto.
6. Administrar proveedores.
7. Generar facturas de venta.
8. Generar facturas de compra.
9. Generar canje de producto
10. Generar transferencia de producto.
11. Despachar producto de hospitalización.
12. Despachar emisión de recetas

13. Pedido de medicamentos a farmacia.
14. Realizar cierres de caja
15. Realizar reportes mensuales
16. Impresiones de reportes mensuales de cualquier índoles que se requiera.
17. Consultas de inventario y kardex

#### **4.3.12. Pruebas de Usabilidad**

Con este tipo de prueba se puede evaluar la utilidad y robustez del sistema, planteando los siguientes parámetros (ver anexo 3):

- ¿Cómo considera la interfaz del sistema?
- ¿Cómo califica las opciones dentro del sistema, organizadas en un solo menú, con etiquetas claramente identificadas?
- ¿Cómo califica los menús, botones y títulos de la aplicación?
- ¿Considera que el software presenta los menús, de acuerdo al tipo de usuario?
- Dentro del uso de la aplicación en general. ¿Cómo califica la navegación en todas las páginas, los colores de los menús, los botones e íconos?
- ¿Cómo califica la pertinencia de los mensajes de información?
- ¿Cómo califica la pertinencia de los mensajes de error?

#### **4.3.13. Pruebas de Seguridad:**

Para el manejo de la seguridad al ingreso del software, se realiza mediante un nombre de usuario y su respectiva clave, además se toma en cuenta el rol de cada usuario donde el sistema presenta las funciones y privilegios correspondientes, Para la seguridad de las claves se usa el código de encriptación bcrypt.

#### **4.3.14. Pruebas de Carga y Estrés**

Las pruebas de carga y estrés tiene la finalidad de verificar que el sistema no tenga ningún inconveniente al momento de ser utilizado. Para estas pruebas se utilizó la herramienta JMETTER la cual es una herramienta libre, en el anexo 8 se encuentra el detalle de la prueba.

#### 4.3.15.Resultados

Como se mencionó anteriormente se realizaron encuestas (Ver anexo 1), para los tipos de pruebas ya citadas, realizando un análisis de las mismas, en la siguiente tabla se muestran los procesos evaluados de acuerdo a las preguntas realizadas a los usuarios.

**MB:** Muy Bueno; **B:** Bueno; **R:** Regular; **M:** Malo.

Preguntas	MB	B	R	M	Total
¿Cómo considera el diseño del sistema?	3	2	0	0	5
¿Cómo considera el tiempo empleado para inicio de sesión?	5	0	0	0	5
¿Cómo califica el menú del sistema?	5	0	0	0	5
¿Cómo califica la información presentada en cada una de las tareas que existen	4	1	0	0	5
¿Cómo califica el ingreso de la información que se requiere para el buen uso de datos del paciente o cliente?	4	1	0	0	5
¿Considera que los formularios de registros son claros?	3	2	0	0	5
¿Cómo califica los menús, botones y títulos de la aplicación?	3	2	0	0	5
¿Considera que los campos de texto de los diferentes formularios son claros como requeridos u obligatorios (*)?	2	3	0	0	5
¿Cómo considera la presentación de las hojas de impresión?	0	5	0	0	5
¿Cómo considera las tablas de consultas?	5	0	0	0	5
¿Considera que las tablas donde están organizados los diferentes clientes, pacientes, productos, proveedores, etc., brindan la búsqueda por criterios (nombre, Historia Clínica, cedula, código, etc.)?	4	1	0	0	5
¿Cómo califica la presentación de tablas, reportes y presentaciones estadísticas?	5	0	0	0	5
¿Cómo califica el menú de acuerdo al tipo de usuario?	0	5	0	0	5
Dentro del uso del sistema en general ¿Cómo califica la navegación en todas las páginas, los colores de los menús, los botones e iconos?	3	2	0	0	5
¿Cómo califica los mensajes de error y aprobación?	4	1	0	0	5
<b>TOTAL</b>	<b>50</b>	<b>25</b>	<b>0</b>	<b>0</b>	<b>75</b>

Con estos totales se procede a calcular los porcentajes con una regla de tres, con su respectiva grafica como lo muestra la Figura 12.

MUY BUENA =  $50 \times 100 \div 75 = 66.67 \%$  BUENA =  $25 \times 100 \div 75 = 33.33 \%$

REGULAR = 0%

MALO = 0 %

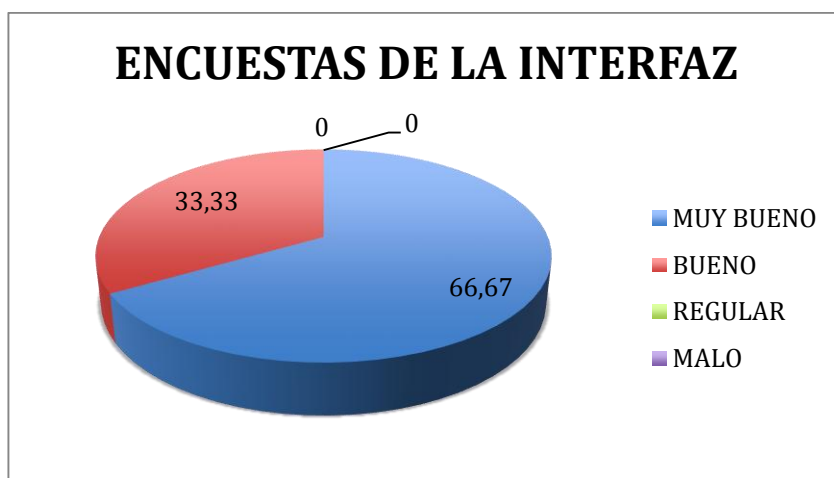


Figura 12. Análisis de encuesta interfaz

Se llega a la conclusión que el 66,67% de los encuestados tiende a decir que la interfaz es muy buena, mientras que el 33,33% dice que la interfaz es buena.

Existió algún problema al realizar las siguientes actividades

	SI	NO
1. Crear un cliente.		5
2. Crear un producto.		5
3. Crear un proveedor.		5
4. Realizar una venta de producto.		5
5. Realizar una compra de producto.		5
6. Realizar un canje de producto.		5
7. Realizar una transferencia de producto		5
8. Realizar un cierre de caja		5
9. Realizar un despacho de medicamento por hospitalización		5
10. Realizar una búsqueda de reportes de cierre de caja		5
11. Realizar una búsqueda de reportes de liquidación		5
12. Editar cliente.		5
13. Editar proveedores		5
14. Editar producto		5
15. Anular transferencia		5
16. Anular despacho de producto por hospitalización		5
17. Anular factura de venta		5
18. Realizar una búsqueda de clientes		5
19. Realizar una búsqueda de proveedor		5
20. Realizar una búsqueda de producto		5
21. Realizar una búsqueda de hospitalización		5
22. Realizar una búsqueda de transferencia de producto		5
23. Realizar el kardex del producto		5
24. Realizar búsqueda de factura de venta		5
25. Realizar búsqueda de factura de compra		5
26. Ver producto caducado		5
27. Anular factura de compra		5
28. Realizar búsqueda de facturas anuladas		5
<b>TOTAL</b>		140



Con estos totales se procede a calcular los porcentajes con una regla de tres, con su respectiva grafica como lo muestra la Figura 13.

$$SI = 0 \times 100 \div 140 = 0 \%$$

$$NO = 140 \times 100 \div 140 = 100 \%$$



Figura 13 Análisis de problemas

Se llega a la conclusión que el 100% de los encuestados tiende a decir que no existe problemas en el sistema.

¿Cómo calificaría la manera de realizar las siguientes actividades?

	Fácil	Normal	Difícil
1. Crear un cliente	3	2	
2. Crear un producto	0	1	
3. Crear un proveedor	0	1	
4. Realizar una venta de producto	3	2	
5. Realizar una compra de producto	0	1	
6. Realizar un canje de producto	0	1	
7. Realizar una transferencia de producto	1	4	
8. Realizar un cierre de caja	3	2	
9. Realizar un despacho de medicamento por hospitalización	2	3	
10. Realizar una búsqueda de reportes de cierre de caja	0	1	
11. Realizar una búsqueda de reportes de liquidación	0	1	
12. Editar un cliente	3	2	
13. Editar un producto	1	0	
14. Editar un proveedor	1	0	
15. Anular transferencia de producto	3	2	
16. Anular despacho de producto por hospitalización	2	3	
17. Anular factura de venta	2	3	
18. Anular factura de compra	1	0	
19. Realizar una búsqueda de clientes	3	2	
20. Realizar una búsqueda de proveedores	1	0	
21. Realizar una búsqueda de productos	3	2	
22. Realizar una búsqueda de hospitalización	3	2	
23. Realizar una búsqueda de transferencia de producto	3	2	
24. Realizar el kardex del producto	0	5	
25. Realizar una búsqueda de factura de venta	2	3	
26. Realizar una búsqueda de factura de compra	0	1	
27. Ver producto caducado	0	1	
28. Realizar búsqueda de facturas anuladas	1	4	
<b>TOTAL</b>	<b>41</b>	<b>51</b>	

Con estos totales se procede a calcular los porcentajes con una regla de tres, con su respectiva grafica como lo muestra la Figura 14.

$$\text{FACIL} = 41 \times 100 \div 92 = 44,56 \% = 45 \%$$

$$\text{NORMAL} = 51 \times 100 \div 92 = 55,44 \% = 55\%$$

$$\text{DIFICIL} = 0\%$$



Figura 14. Análisis del grado de dificultad del sistema.

Se llega a la conclusión que el 55% de los encuestados tiende a decir que la dificultad del sistema es normal, mientras que el 45 % tiende a decir que fácil.

### **Conclusión final.**

Se ha llegado a la conclusión que el 75 % de la población hace uso de buena manera de el sistema sin complicaciones de ninguna medida, dando así solución a una de las metas planteadas que es el de dar soluciones a necesidades que se presentaron durante el transcurso del desarrollo.

## **g. Discusión**

### **1. Desarrollo de la propuesta alternativa**

La presente tesis denominada **“Sistema web de gestión de farmacia, basado en tecnología móvil para el Hospital básico 7 BI Loja”**, proporcionó a esta institución la implantación de un sistema integrado web que permita la comunicación directa entre los servicios de farmacia, estadística y central de enfermería, con la finalidad de que la sociedad en general se beneficie y sea atendida de mejor manera bajo la misión y visión del hospital.

El objetivo general de este proyecto es logrado en su totalidad, en cuanto a lo que dice el objetivo general en base a que el proyecto estaba para realizarse en tecnología móvil, se tuvo que dirigir y desarrollar en el framework de desarrollo Ruby on Rails, como el Framework Bootstrap ya que las autoridades del hospital decidieron no implantarlo para dispositivos móviles por la consecuencia que llevaba a esta institución a desembolsar cantidad mayores al presupuesto que tiene estipulado por año, es así que se realizó y se diseñó en Bootstrap para que tenga coherencia y semejanza en su objetivo, es por ello que hago conocer el certificado del director de la institución para respaldar mi objetivo (Ver anexo 4,5).

A continuación se detalla los objetivos específicos y como se logró su total cumplimiento:

#### **1. Elaborar el módulo de farmacia usando el framework de desarrollo Ruby on Rails para la gestión de información.**

Para llevar a cabo este objetivo se ha hecho uso de las fases de la metodología XP. (Ver sección de resultados, Fases 1, Planificación del proyecto), cumpliendo con las siguientes actividades:

1. Elaboración de especificaciones de lo que va a realizar el sistema de farmacia.
2. Requerimientos funcionales y no funcionales del sistema.
3. Lista de los actores del sistema.
4. Características de los usuarios.
5. Historias de usuario

## **2. Desarrollo del inventario de productos de farmacia y kardex.**

Para llevar a cabo este objetivo se ha hecho uso de las fases de la metodología XP. (Ver sección de resultados, Fases 2, diseño), cumpliendo con las siguientes actividades:

1. Realización de las tarjetas CRC.
2. Diseño simple de la interfaz de producto y su respectivo kardex

## **3. Acoplar el sistema para dispositivos móviles usando el framework de desarrollo responsive Bootstrap.**

Para llevar a cabo este objetivo se ha hecho uso del capítulo 3 apartado 3.4 cumpliendo con las siguientes actividades:

1. Carga de Javascript a la ruta:  
vendedor/ascetas/javascript/bootstrap.js
2. Carga de css en la ruta :  
vendedor/ascetas/stylesheets/bootstrap.css

A continuación en la Figura 15 se indica como es la carga de lo antes mencionado.

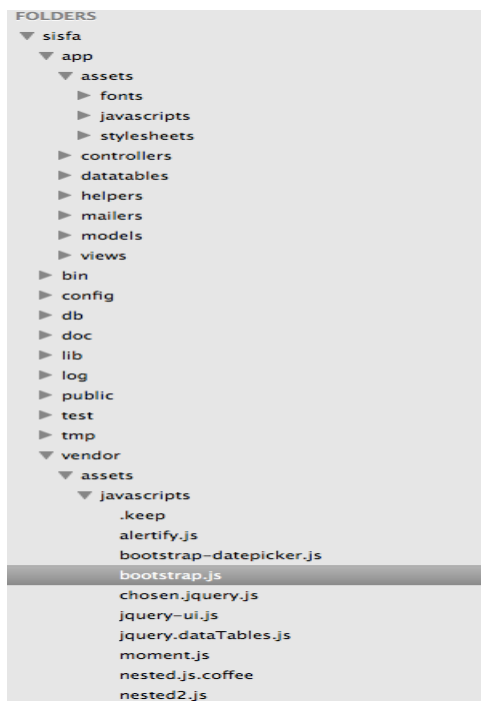


Figura 15. Estructura de archivos para la carga de Bootstrap

La visualización de esto se refleja en la Figura 16 que indica como se ve desde una netbook de 10'.

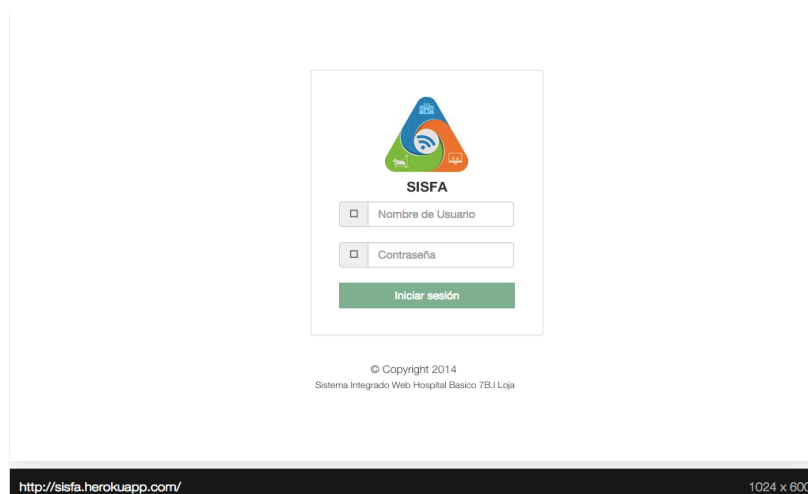


Figura 16. Bootstrap en notebook

La visualización de esto se refleja en la Figura 17 que indica como se ve desde una ipad.

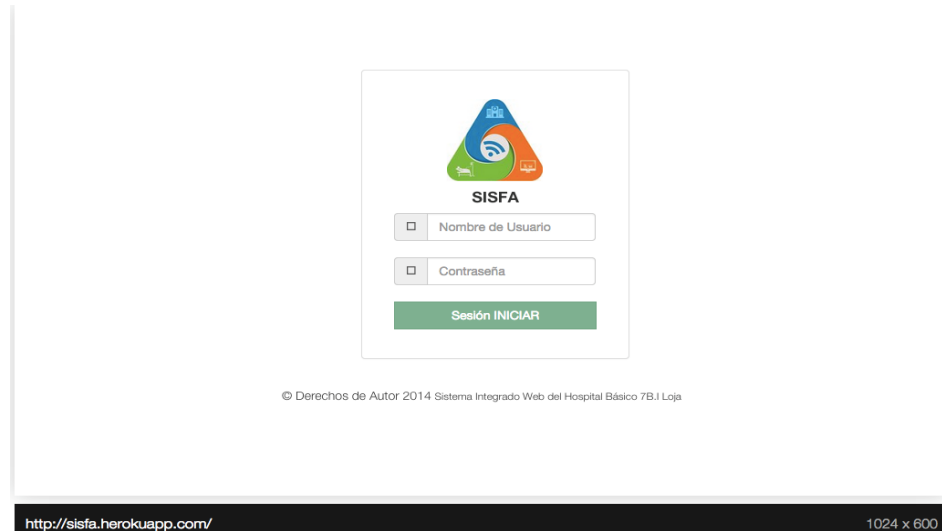


Figura 17. Bootstrap en una ipad

La visualización de esto se refleja en la Figura 18 que indica como se ve desde un celular galaxy s3.



Figura 18. Bootstrap en un galaxy s3

## **2. Valoración técnica económica ambiental**

El desarrollo de este software se lo elaboró tomando en cuenta que la institución como es el Hospital Básico 7 BI Loja, es una entidad de renombre nacional como local y se vio la necesidad de impartir conocimientos y ayudar de formar conjunta en la realización de un sistema integrado de diferentes dependencias, para agilizar procesos que se manejan de forma diaria, es así que se ha creído conveniente justificar este punto en los siguientes ámbitos:

### **2.1. Valoración técnica**

Para la realización del presente proyecto se cuenta con medios tecnológicos y herramientas (hardware, software libre, etc.) necesarias que se usará a lo largo del desarrollo del proyecto. Además es factible de realizar porque se cuenta con bibliografía necesaria en nuestra misma Universidad la cual ayudará a sustentar y a poder realizar de manera eficaz el proyecto.

Dentro del entorno técnico se justifica debido a la necesidad de su uso que se plantea en el presente y futuro, luego de dialogar con las autoridades del Hospital Militar, llegando a explicar las ventajas que trae consigo la implementación de este tipo de sistema, ellos nos brindaran la información necesaria así como el personal que labora en el hospital.

### **2.2. Valoración económica.**

Económicamente se cuenta con el presupuesto y tiempo por parte del desarrollador del proyecto como también el de la institución para poder adquirir equipos para solventar el desarrollo de este proyecto. Tomando en consideración que este proyecto será de gran aporte a la solución de los problemas existentes en el Hospital Militar y que a la vez permite cumplir a cabalidad con los objetivos planteados, a continuación se detalla los recursos humanos, económicos, materiales, técnicos y tecnológicos, con sus costes respectivos.



### 2.2.1. Talento Humano

Muestra un estimado del tiempo y costo asignado al investigador y tutor responsables de la terminación del proyecto.

TABLA LXXIV : TALENTO HUMANO

Descripción	Cantidad	Horas	Valor/unit	Valor/Total
Investigador	1	960	\$3.00	\$2880.00
Asesor de Proyecto	1	240	\$5.00	\$1200.00
Director de Tesis	1	0	\$0.00	\$0.00

### 2.2.2. Bienes

A continuación el estimado del presupuesto Tecnológico de hardware y software que se requieren para la elaboración del presente proyecto.

#### Presupuesto Tecnológico.

#### Presupuesto Materiales

TABLA LXXV : MATERIALES

Cant	Materiales	Valor / Unit	Valor / Total
1000	Hojas A4	\$ 0.15	\$ 15.00
3	Anillado	\$ 2.00	\$ 6.00
8	Materiales de oficina	\$ 1.00	\$ 8.00
4	Cartuchos de impresora Canon	\$22	\$88.00
3	Borde o Perfil	\$ 0.50	\$ 1.50
5	CD's	\$ 0.80	\$ 4.00

TABLA LXXVI : HARDWARE Y SOFTWARE

Cant	Descripción de Equipos Técnicos	Tiempo Utilización (Mes)	Valor/ unit	Tiempo de Vida (Año)	Valor/ Total
	<b>Hardware</b>				
1	Computador Portátil Toshiba satélite L635	12	\$ 1200	5	\$ 240.00
1	Impresora Canon Mp190	12	\$ 80	3	\$ 27
1	Servidor hp profile	5	\$ 600.00		\$ 600.00
	<b>Software</b>				
1	Ubuntu server 12.04 LTS	1	\$ 0.00		\$ 0.00
1	Ruby on Rails	1	\$ 0.00		\$ 0.00

### 2.2.3. Servicios

Son los servicio que se utilizara a lo largo del desarrollo, indicando precios reales que implican el uso de dichos recursos. Cabe mencionar que la capacitación virtual son videos tutoriales pagados, que se desarrollan en línea.

TABLA LXXVII : SERVICIOS

Servicio	Descripción	Valor/unit	Valor/Total
Transporte	250	\$ 0.25	\$ 62.50
Internet	12 meses	\$ 20	\$ 240
Capacitación Virtual	Video Tutoriales Lynda.com	\$ 500	\$ 500

### 2.2.4. Resumen del Presupuesto

Se presenta el presupuesto total de las tablas ya descritas anteriormente y a su vez un estimado de un 4% de imprevistos del total de gastos.

TABLA LXXVIII : RESUMEN

Resumen del Presupuesto	Costo Total
Talento Humano	\$ 4080.00
Bienes	\$ 989.50
Servicios	\$ 802.50
SUBTOTAL	\$ 5872.00
Imprevistos 4 %	\$ 234.88
TOTAL	\$ 6106,88

### **2.3. Valoración ambiental**

En cuanto a este punto el impacto ambiental que se da es la disminución de papel de oficina, por reportes o información que se encuentran en el sistema, así como también el servidor que trabaja la institución es un servidor en la nube que será una herramienta que por ser de una empresa internacional su cuidado y seguridad es muy alta.

## **h. Conclusiones.**

Luego de haber cumplido con los objetivos planteados al inicio del proyecto y de haber culminado con éxito el PFC; se ha llegado a las siguientes conclusiones.

1. La utilización de registros en el sistema permite al personal de la institución optimizar el desenvolvimiento de trabajo y con ello mejorar el servicio al cliente.
2. La metodología XP es la idónea y de mejor acoplamiento a sistemas web, pues siempre se involucra al usuario final; y debido a sus etapas: planificación, diseño, desarrollo y pruebas, se obtiene la correcta funcionalidad del sistema.
3. La planificación del sistema permitió determinar los requerimientos del usuario, constituyéndose en el pilar principal para el proceso de las etapas y con ello se pudo cumplir las expectativas inicialmente planteadas para así ofrecer el sistema web que hoy en día ya se ha implantado.
4. La validación de usuarios ayuda a detectar errores durante el desarrollo del sistema; con el fin de brindar la aceptación y validación que se debe obtener para el manejo adecuado del mismo.
5. Las herramientas de desarrollo utilizadas, fueron determinantes en el diseño del sistema ya que así se logró un fácil manejo del mismo para el usuario en cuanto a las tareas a realizar.

## **1. Recomendaciones**

1. Seleccionar siempre la metodología adecuada en este tipo de sistemas web para así evitar problemas en la codificación del sistema.
2. Trabajar siempre en base a las necesidades del usuario final, puesto que es el único que manejará el sistema, y su aporte contribuirá a que en un ciento por ciento exista un correcto funcionamiento del mismo.
3. Utilizar herramientas adecuadas que permitan una integración de componentes, lenguajes con la finalidad de garantizar el correcto diseño del sistema.
4. Realizar pruebas de aceptación, validación y usabilidad, necesarias para asegurar que el sistema web desarrollado funcione de manera adecuada; y así evitar cualquier tipo de error.
5. Se recomienda al usuario que maneja el sistema no compartir contraseñas a personal no autorizado, esto con el fin de evitar pérdidas o alteración de información, tanto de productos, clientes, proveedores o reportes financieros.
6. Se recomienda a la institución no implementar la aplicación móvil del sistema, puesto que la información que se maneja es confidencial; y en base a las tareas que realiza el personal no es necesario este tipo de aplicaciones.

## **j. Bibliografía**

[1] Ruby on Rails [En línea]: <<https://www.ruby-lang.org/es/about/>> [Consultado el 14 de Septiembre de 2013].

[2] Ruby on Rails como herramienta de desarrollo [En Línea]: <<http://aspgems.com/blog/leticia/webinar-por-que-ruby-rails>> [Consultado el 16 de diciembre de 2013].

[3] Pasos para instalar ciento por ciento seguro Rails en mac [En Línea]: <<http://railsapps.github.io/installrubyonrailsmac.html>> [Consultado el 16 de diciembre de 2013].

[4] Arquitectura de una aplicación desarrollada con RoR. [En línea]: <[http://www.windarooci.com/assets/service/main\\_image\\_url/1/rails\\_architecture.png](http://www.windarooci.com/assets/service/main_image_url/1/rails_architecture.png)> [Consultado el 15 de Septiembre de 2013].

[5] FERNANDEZ Gerardo. Introducción a Extreme Programming [En línea]: <<http://aalbertovargas.files.wordpress.com/2011/07/presentacionxp.pdf>> [Consultado el 10 de Febrero de 2013].

[6] José H. Canós, Patricio Letelier y M<sup>a</sup> Carmen Penadés, Metodologías Ágiles en el Desarrollo de Software. [En línea]: <[http://noqualityinside.com.ar/nqi/nqifiles/XP\\_Agil.pdf](http://noqualityinside.com.ar/nqi/nqifiles/XP_Agil.pdf)>. [Consultado 3 Marzo del 2014].

[7] Echeverry Tobón, Luís Miguel, Delgado Carmona, Luz Elena, Caso práctico de la metodología ágil XP al desarrollo de software.

<<http://recursosbiblioteca.utp.edu.co/dspace/handle/11059/794>>  
[Consultado 3 Marzo del 2014].

[8] Universidad del Valle. Cali. Colombia. Programación Extrema.<<http://eisc.univalle.edu.co/materias/WWW/material/lecturas/xp.pdf>>[Consultado 4 Julio del 2014].

[9] Flanagan, David, JavaScript: The Definitive Guide (4<sup>a</sup> Edición edición) (2002).

[10] Miguel Angel Álvarez, Introducción a JQuery <http://www.desarrolloweb.com/articulos/introduccion-jquery.html>  
[Consultado 4 Julio 2014].

[11] J Ashkenas, The Little Book on CoffeeScript [http://jameslavin.com/Little\\_Book\\_on\\_CoffeeScript.pdf](http://jameslavin.com/Little_Book_on_CoffeeScript.pdf), [Consultado Agosto 2014].

[12] Javier Eguiluz, Introducción a CSS, <http://librosweb.es/css/>,  
[Consultado 19 de Agosto 2014].

[13] Mark Otto, Jacob Thornton, Bootstrap 3 el manual oficial [http://librosweb.es/bootstrap\\_3](http://librosweb.es/bootstrap_3), [Consultado 19 de Agosto 2014].

[14] MySQL, características En línea]:  
<<http://dev.mysql.com/doc/refman/5.0/es/features.html>> [Consultado el 14 de Septiembre de 2013].



## k. Anexos

### Anexo 1: Encuestas realizadas a la administradora y auxiliares de farmacia sobre la interfaz del sistema.

Esta encuesta nos permitió conocer la aceptación de la interfaz del sistema, para poder comprender en qué nivel de diseño se cumplió con lo requerido por la licenciada.

Preguntas	MB	B	R	M	Total
¿Cómo considera el diseño del sistema?					
¿Cómo considera el tiempo empleado para inicio de sesión?					
¿Cómo califica el menú del sistema?					
¿Cómo califica la información presentada en cada una de las tareas que existen					
¿Cómo califica el ingreso de la información que se requiere para el buen uso de datos del paciente o cliente?					
¿Considera que los formularios de registros son claros?					
¿Cómo califica los menús, botones y títulos de la aplicación?					
¿Considera que los campos de texto de los diferentes formularios son claros como requeridos u obligatorios (*)?					
¿Cómo considera la presentación de las hojas de impresión?					
¿Cómo considera las tablas de consultas?					
¿Considera que las tablas donde están organizados los diferentes clientes, pacientes, productos, proveedores, etc., brindan la búsqueda por criterios (nombre, Historia Clínica, cedula, código, etc.)?					
¿Cómo califica la presentación de tablas, reportes y presentaciones estadísticas?					
¿Cómo califica el menú de acuerdo al tipo de usuario?					
Dentro del uso del sistema en general ¿Cómo califica la navegación en todas las páginas, los colores de los menús, los botones e iconos?					
¿Cómo califica los mensajes de error y aprobación?					
<b>TOTAL</b>					

## Anexo 2: Encuestas realizadas a la administradora y auxiliares de Farmacia sobre problemas que puedan existir en el sistema.

Esta encuesta nos permitió conocer si existe algún problema al momento de realizar alguna actividad.

	SI	NO
1. Crear un cliente.		
2. Crear un producto.		
3. Crear un proveedor.		
4. Realizar una venta de producto.		
5. Realizar una compra de producto.		
6. Realizar un canje de producto.		
7. Realizar una transferencia de producto		
8. Realizar un cierre de caja		
9. Realizar un despacho de medicamento por hospitalización		
10. Realizar una búsqueda de reportes de cierre de caja		
11. Realizar una búsqueda de reportes de liquidación		
12. Editar cliente.		
13. Editar proveedores		
14. Editar producto		
15. Anular transferencia		
16. Anular despacho de producto por hospitalización		
17. Anular factura de venta		
18. Realizar una búsqueda de clientes		
19. Realizar una búsqueda de proveedor		
20. Realizar una búsqueda de producto		
21. Realizar una búsqueda de hospitalización		
22. Realizar una búsqueda de transferencia de producto		
23. Realizar el kardex del producto		
24. Realizar búsqueda de factura de venta		
25. Realizar búsqueda de factura de compra		
26. Ver producto caducado		
27. Anular factura de compra		
28. Realizar búsqueda de facturas anuladas		
<b>TOTAL</b>		

### Anexo 3: Encuestas realizadas a la administradora y auxiliares de Farmacia sobre el grado de dificultad que existen al realizar una actividad o tarea.

Esta encuesta nos permitió conocer el grado de dificultad del usuario para realizar tareas que realiza el sistema como son crear producto, realizar una factura, etc.

	Fácil	Normal	Difícil
1. Crear un cliente			
2. Crear un producto			
3. Crear un proveedor			
4. Realizar una venta de producto			
5. Realizar una compra de producto			
6. Realizar un canje de producto			
7. Realizar una transferencia de producto			
8. Realizar un cierre de caja			
9. Realizar un despacho de medicamento por hospitalización			
10. Realizar una búsqueda de reportes de cierre de caja			
11. Realizar una búsqueda de reportes de liquidación			
12. Editar un cliente			
13. Editar un producto			
14. Editar un proveedor			
15. Anular transferencia de producto			
16. Anular despacho de producto por hospitalización			
17. Anular factura de venta			
18. Anular factura de compra			
19. Realizar una búsqueda de clientes			
20. Realizar una búsqueda de proveedores			
21. Realizar una búsqueda de productos			
22. Realizar una búsqueda de hospitalización			
23. Realizar una búsqueda de transferencia de producto			
24. Realizar el kardex del producto			
25. Realizar una búsqueda de factura de venta			
26. Realizar una búsqueda de factura de compra			
27. Ver producto caducado			
28. Realizar búsqueda de facturas anuladas			
<b>TOTAL</b>			

**Anexo 4: Informe del costo que lleva implantar el sistema para dispositivos móviles presentado por el ing. Juan Pablo Sánchez Oviedo al Crol. Juan Carlos Salguero Director del Hospital.**

INFORME QUE PRESENTA EL ING. JUAN PABLO SÁNCHEZ AL CRNL. JUAN CARLOS SALGUERO S. DIRECTOR DEL HOSPITAL BASICO 7 BI LOJA ACERCA EL COSTO DEL LA IMPLEMENTACION DEL SISTEMA INFORMATICO SISFA PARA DISPOSITIVOS MOVILES.

LOJA, 21 DE JULIO DE 2014

**ANTECEDENTES.**

Como encargado del área de sistemas de esta casa de salud me permito informar a usted Cml acerca del costo que se representaría la implementación del sistema **SISFA**. En dispositivos móviles

**DESARROLLO.**

El sistema que actualmente se encuentra en desarrollo para la integración de los servicios de estadística, farmacia y central de enfermería también se pensó en desarrollar para dispositivos móvil pero para ello se debe contar con los siguientes recursos para su implantación:

- Contar con un servidor web para el alojamiento del sistema y poder acceder desde cualquier lugar.
- Contar con un servicio para el alojamiento de la aplicación móvil desarrollada y sería dependiendo del sistema operativo con que cuenta los diferentes móviles o Tablet de la institución, es decir IOS. Android, Windows Phone, etc.
- Contar con la infraestructura para que todos puedan acceder de forma wifi a el servidor donde estaría alojada la aplicación.

El beneficio sería que es una aplicación que traerá ayuda a esta institución pero sabiendo que la institución no cuenta con los recursos económicos para este tipo sistema en versión móvil quedaría desarrollada de forma innecesaria puesto que el costo total sobre pasaría unos 10.000 dólares en adquirir el hardware y montar la infraestructura que se necesita para dicha aplicación.

## CONCLUSIONES:

- Costo en adquirir e implantar dicha aplicación muy excesivos
- Sistema que no se lo utilizara continuamente de forma inalámbrica si no con conexión cableada.

## RECOMENDACIONES.

Es por ello me dirijo a usted para que se autorice en la no realización de este sistema SISFA en versión móvil puesto que los costos antes mencionados no se los posee en la actualidad.

Esto es cuanto puedo informar en honor a la verdad

Atentamente.

Ing. Juan Pablo Sánchez

RUC: 11043695-1  
CI: 11043695  
TÉCNICO DEL HB-7 B1



SECCION:

FECHA: 22-04-2014

HORA: 11:00

ACCION TOMADA:

**Anexo 5: Certificado de NO realizar la aplicación móvil  
certificada por el Director del hospital.**



**HOSPITAL BASICO 7 BI LOJA**

**CRNL. EM  
JUAN C. SALGUERO S.  
DIRECTOR DEL HB 7 BI "LOJA".**

**CERTIFICA:**

Luego de revisar la sugerencias presentadas por el sr. ING. JUAN PABLO SANCHEZ OVIEDO, con cédula 1104369515, acerca de la implantación de un sistema que actualmente se encuentra en desarrollo tipo web y misma que se desarrollara en versión móvil por parte de los señores tesisistas: ANGEL ALBERTO VALDEZ MASACHE CON CI: 1103786990 Y LENIN ISRAEL CAPA MEDINA con cedula de identidad 1104706922, mismo que luego de las debidas averiguaciones acerca del valor económico que traerá consigo la implantación de dicha aplicación y sabiendo que la institución no cuenta con los recursos económicos presupuestados certifico que no se implemente ese tipo versión en esta casa de salud.

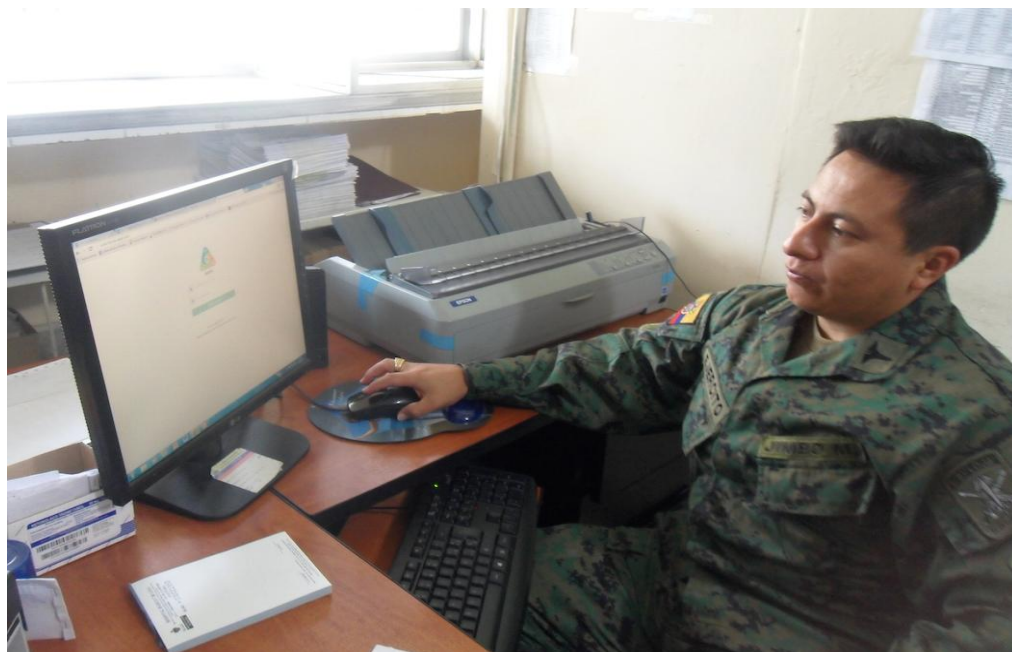
Es todo cuanto puedo certificar, documento que servirá únicamente para lo solicitado.

Loja, 22 de Julio de 2014

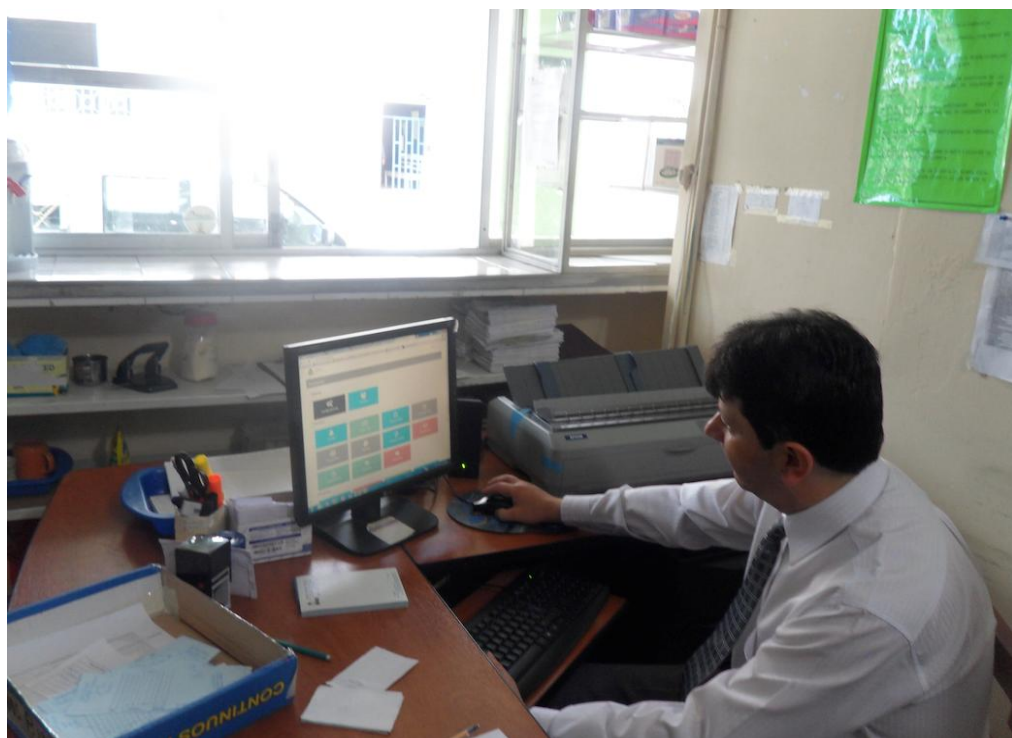
  
  
**CRNL. EM  
JUAN C. SALGUERO S.  
DIRECTOR DEL HB 7 BI "LOJA".**



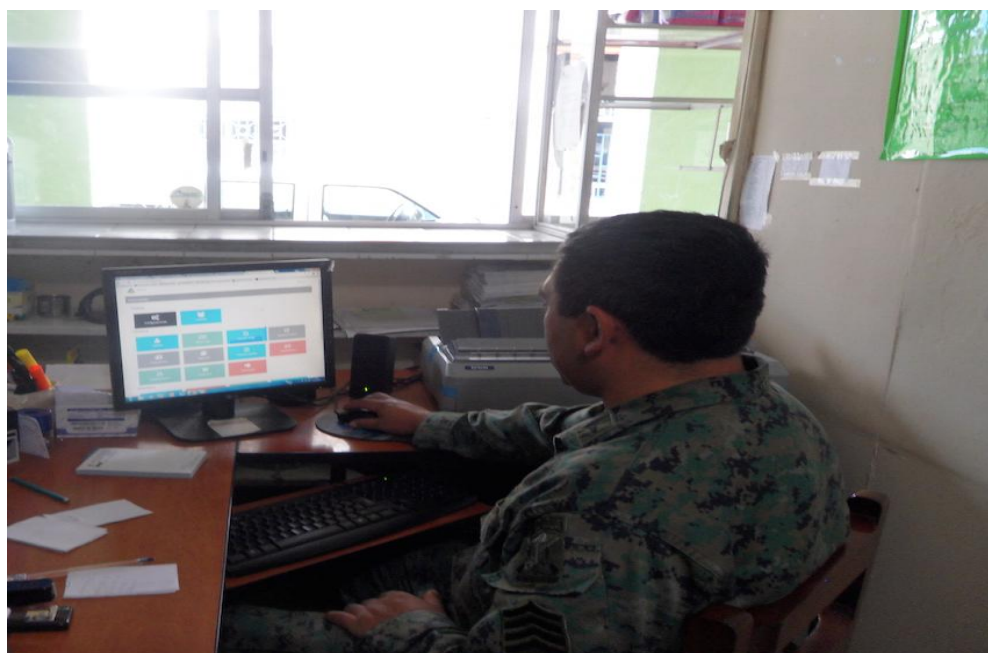
## Anexo 6: Fotos de personal de farmacia.



CboP. Marcelo Jimbo Auxiliar de Farmacia



Dr. Flavio Calvache Auxiliar de Farmacia



SgtoP. Jorge Cueva Auxiliar de Farmacia



## Anexo 7: Validación HTML 5.



← → validator.w3.org

Esta página se ha traducido del inglés al español. [Mostrar original](#) [Configuración](#)

**W3C** Servicio de Validación de marcado  
Compruebe el marcado (HTML, XHTML, ...) de los documentos Web

Validar por URI   Validar por carga de archivos   Validar mediante entrada directa

Validar por URI

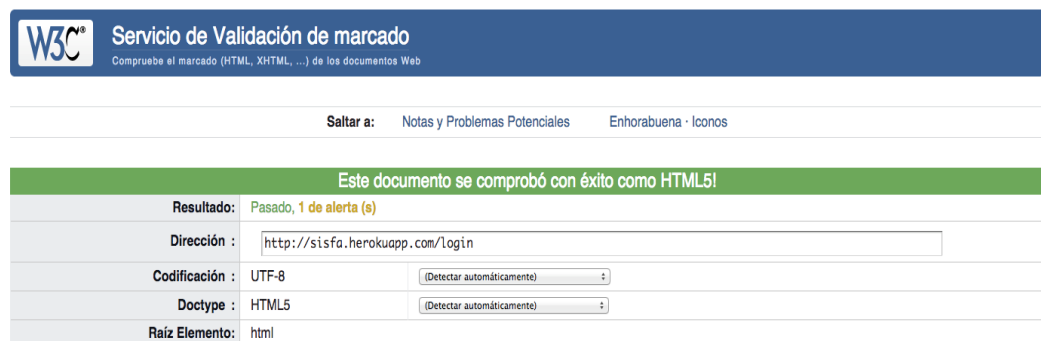
Validar un documento en línea:

Dirección:

► Más opciones

Comprobar

Página en internet para realizar prueba de HTML del proyecto



**W3C** Servicio de Validación de marcado  
Compruebe el marcado (HTML, XHTML, ...) de los documentos Web

Saltar a: [Notas y Problemas Potenciales](#) [Enhorabuena · Iconos](#)

Este documento se comprobó con éxito como HTML5!

Resultado:	Pasado, 1 de alerta (s)	
Dirección :	<input type="text" value="http://sisfa.herokuapp.com/login"/>	
Codificación :	UTF-8	<input type="button" value="(Detectar automáticamente)"/>
Doctype :	HTML5	<input type="button" value="(Detectar automáticamente)"/>
Raíz Elemento:	html	

Respuesta de la validación del HTML.

## Anexo 8: Pruebas JMeter.

### 1. Pruebas de Carga, Rendimiento y Estrés

Las pruebas de carga, rendimiento y estrés fueron realizadas por los desarrolladores del proyecto, permitiendo verificar el correcto funcionamiento de la aplicación cuando existen accesos simultáneos de usuarios.

Para este tipo de pruebas se ha utilizado la herramienta Apache JMeter™ que es una aplicación de escritorio de código abierto, desarrollada en Java.

Estas pruebas se las ha realizado contra el servidor de Heroku, de características básicas:

1. 1X CPU Share
2. RAM: 512MB
3. Swap space: 1GB max
4. Storage space: 100MB max

#### 1. Prueba de Carga

Las pruebas de carga tienen la finalidad de medir la capacidad máxima que tiene un servidor web (hardware y software), simulando varios usuarios que realizan una petición a la aplicación al mismo tiempo.

En la imagen se muestra las pruebas realizadas al módulo de farmacia con 5 usuarios realizando la misma petición a la vez, el margen de error es de 0%.

Label	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error
Pagina de Inicio	5	620	439	1339	304	1339	0,00%
Autenticacion de usuario	5	1843	1940	2095	1243	2095	0,00%
Factura de Venta	5	709	586	952	541	952	0,00%
Factura Anulada	5	580	589	629	518	629	0,00%
Facturar Receta	5	637	664	685	563	685	0,00%
Transferencia de producto	5	655	651	705	593	705	0,00%
Transferencia anulada	5	596	594	632	571	632	0,00%
Crear factura de compra	5	737	693	899	618	899	0,00%
Canje por otro producto	5	605	582	709	512	709	0,00%
Crear Proveedor	5	583	595	628	531	628	0,00%
Canje de producto por el mismo	5	606	594	653	562	653	0,00%
Pedido medicamento anulado	5	603	597	722	522	722	0,00%
Despachar producto	5	665	666	802	569	802	0,00%
Reporte de venta	5	671	713	744	582	744	0,00%
Reporte Liquidacion	5	614	586	708	531	708	0,00%
Reporte cierre de caja mensual	5	584	591	729	481	729	0,00%
Reporte cierre caja diario	5	578	560	667	542	667	0,00%
Reporte de Inventario	5	553	542	617	497	617	0,00%
Productos caducados	5	573	568	638	529	638	0,00%
Editar producto	5	588	595	708	493	708	0,00%
Kardex Producto	5	636	627	736	576	736	0,00%
Logout	5	596	547	926	367	926	0,00%
TOTAL	110	674	597	845	304	2095	0,00%

Resultado de la prueba de carga del módulo de farmacia.

Se estima una crecimiento de 5 usuarios más en los próximos 3 años, esto en base a las nuevas instalaciones de la institución, por lo cual se realiza las pruebas con 10 usuarios al mismo tiempo, los resultados se aprecian en la imagen.

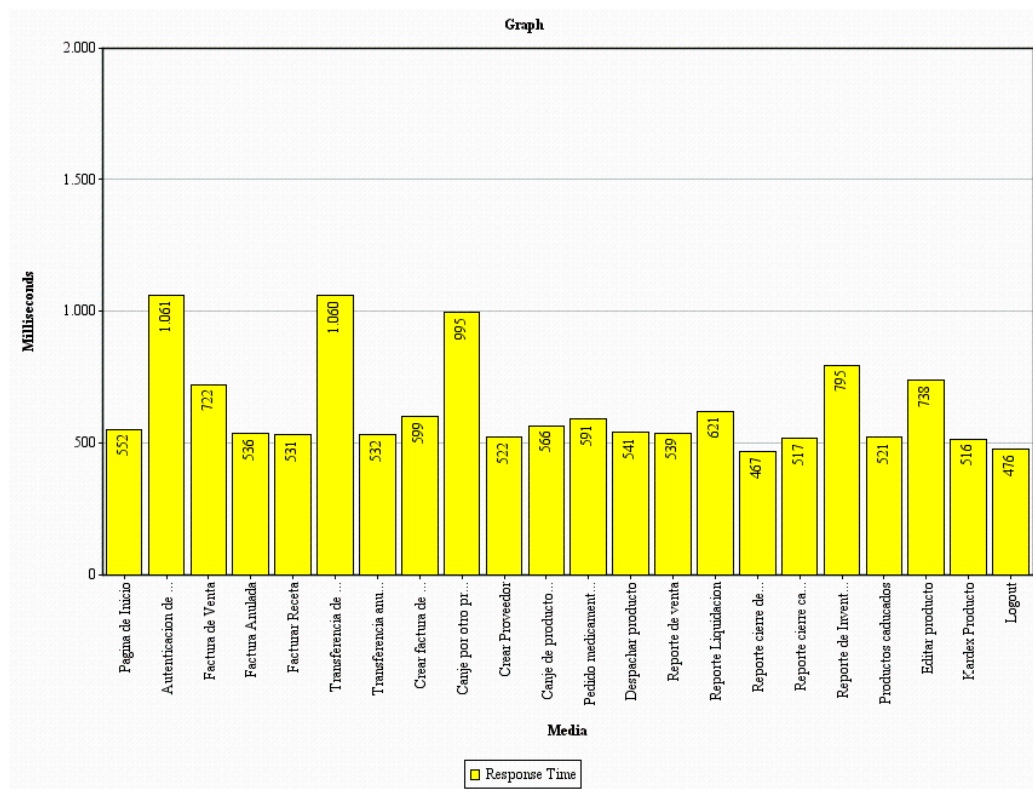
Label	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error
Pagina de Inicio	10	816	566	1633	319	1633	0,00%
Autenticacion de usuario	10	2768	2980	3554	1787	3554	0,00%
Factura de Venta	10	776	721	1568	564	1568	0,00%
Factura Anulada	10	656	614	823	534	823	0,00%
Facturar Receta	10	729	783	908	538	908	0,00%
Transferencia de producto	10	680	676	924	399	924	0,00%
Transferencia anulada	10	716	719	943	586	943	0,00%
Crear factura de compra	10	877	859	1070	760	1070	0,00%
Canje por otro producto	10	773	724	1066	652	1066	0,00%
Crear Proveedor	10	702	686	976	581	976	0,00%
Canje de producto por el mismo	10	647	623	792	583	792	0,00%
Pedido medicamento anulado	10	665	698	783	502	783	0,00%
Despachar producto	10	643	628	729	564	729	0,00%
Reporte de venta	10	663	691	828	495	828	0,00%
Reporte Liquidacion	10	616	626	713	542	713	0,00%
Reporte cierre de caja mensual	10	668	649	804	580	804	0,00%
Reporte cierre caja diario	10	656	614	811	563	811	0,00%
Reporte de Inventario	10	764	760	1086	556	1086	0,00%
Productos caducados	10	682	733	794	541	794	0,00%
Editar producto	10	893	593	3540	549	3540	0,00%
Kardex Producto	10	566	601	783	408	783	0,00%
Logout	10	492	506	644	371	644	0,00%
TOTAL	220	793	669	943	319	3554	0,00%

Resultado de la prueba de carga del módulo de farmacia con proyección a 10 usuarios.

## 2. Prueba de Rendimiento

Las pruebas de rendimiento son utilizadas para medir el tiempo de respuesta de una cierta tarea.

En la imagen se visualiza el tiempo de respuesta en milisegundos que demoran en responder cada una de las operaciones que se realizan en el módulo de farmacia el rendimiento mínimo es 467ms, el rendimiento máximo es de 1061ms y el rendimiento promedio de estas actividades es de 637ms.



Resultado de la prueba de rendimiento del módulo de estadística.

### 3. Prueba de estrés

Las pruebas de estrés son utilizadas para determinar cuándo una aplicación es incapaz de responder a las peticiones, en la imagen se muestra que si realizan 100 peticiones por segundo, el sistema no es capaz de responder a todas las solicitudes, dando un margen de error del 1% en la parte de Factura de venta.

Label	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error
Página de Inicio	100	4393	5999	7273	372	7392	0.00%
Autenticación de usuario	100	20707	20078	23091	18448	23721	0.00%
Factura de Venta	100	6428	5605	9225	3238	37978	1.00%
Factura Anulada	100	4301	4009	4670	3431	11647	0.00%
Facturar Receta	100	5550	4284	7647	3371	27250	0.00%
Transferencia de producto	100	7630	4869	11568	3447	49355	1.00%
Transferencia anulada	100	4200	3868	5207	3253	13128	0.00%
Crear factura de compra	100	7227	5675	10481	4787	17359	0.00%
Canje por otro producto	100	6033	5288	8656	4788	11893	0.00%
Crear Proveedor	100	6365	5354	8893	3516	18938	0.00%
Canje de producto por el mismo	100	6918	5918	9347	4944	17294	0.00%
Pedido medicamento anulado	100	5978	4384	10860	3357	32657	1.00%
Despachar producto	100	5839	4926	8789	3459	28374	0.00%
Reporte de venta	100	6781	5344	9201	2802	58206	0.00%
Reporte Liquidación	100	6253	5839	8627	1190	17084	0.00%
Reporte cierre de caja mensual	100	6868	5933	9568	1620	28181	0.00%
Reporte cierre caja diario	100	6565	6111	9479	871	29990	0.00%
Reporte de Inventario	100	5737	5974	6821	450	9697	0.00%
Productos caducados	100	5836	6061	7128	828	28039	0.00%
Editar producto	100	5680	5654	12148	375	13386	0.00%
Kardex Producto	100	4375	4596	6347	464	9284	0.00%
Logout	100	3777	4131	6302	387	10513	0.00%
TOTAL	2200	6520	5397	9795	372	58206	0.14%

Resultado de la prueba de estrés

## Anexo 9: Certificación de traducción del resumen.

CENTRO DE ENSEÑANZA DE IDIOMA INGLÉS  
FINE-TUNED ZAMORA

Zamora, 8 de diciembre de 2014

Lic. Erika Tapia

**COORDINADORA DE FINE TUNED ENGLISH-ZAMORA**

CERTIFICA:

A petición del egresado ANGEL ALBERTO VALDEZ MASACHE , se realizo la traducción del resumen del trabajo de titulación con el tema: **SISTEMA WEB DE GESTION DE FARMACIAS BASADO EN TECNOLOGÍA MÓVIL PARA EL HOSPITAL BÁSICO 7B.I "LOJA"**.

Pudiendo hacer uso de este certificado en lo que se requiera conveniente

Atentamente

Lic. Erika Tapia  
COORDINADORA FINE TUNED ZAMORA

**Anexo 10: Certificación de haber implantado el sistema  
en el Hospital Básico 7BI - Loja.**



## HOSPITAL BASICO 7 BI LOJA

**CRNL. DE EM  
JUAN C. SALGUERO S.  
DIRECTOR**

**CERTIFICA:**

**Que el SR. VALDEZ MASACHE ANGEL ALBERTO, con cédula 1103786990, realizó su tema de tesis dominado "SISTEMA WEB DE GESTION DE FARMACIA, BASADO EN TECNOLOGIA MOVIL PARA EL HOSPITAL BASICO 7 BI "LOJA" trabajo que culminó con disciplina y responsabilidad.**

**Es todo cuanto puedo certificar, documento que servirá únicamente para lo solicitado.**

Loja, 06 de noviembre de 2014.



**JUAN C. SALGUERO S.  
CRNL. DE EM.  
DIRECTOR DEL HB-7 BI LOJA**



## Anexo 11: Licencias

En este trabajo de titulación se ha aplicado dos tipos de licencia, uno para el código fuente de la aplicación MACOOL y otro tipo de licencia para el presente documento y los manuales de usuario y programador. A continuación se detalla cada tipo:

La licencia Apache es una licencia de software libre creada por la Apache Software Foundation (ASF). La licencia Apache requiere la conservación del aviso de copyright y el disclaimer, pero no es una licencia copyleft, ya que no requiere la redistribución del código fuente cuando se distribuyen versiones modificadas.

Como cualquier otra de las licencias de software libre, la Licencia Apache permite al usuario del software la libertad de usarlo para cualquier propósito, distribuirlo, modificarlo, y distribuir versiones modificadas de ese software.

**APPENDIX:** How to apply the Apache License to your work. To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

**Copyright {2014} {A.Valdez, L.Capa}**

Licensed under the Apache License, Version 2.0 (the "License");  
You may not use this file except in compliance with the License.  
You may obtain a copy of the License at





<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## LICENCIA CREATIVE COMMONS

Atribución-No Comercial-Compartir igual CC BY-NC-SA

Esta licencia permite a otros distribuir, re mezclar, retocar, y crear a partir de tu obra de modo no comercial, siempre y cuando den crédito y licencien sus nuevas creaciones bajo las mismas condiciones.

- |   |                        |  |
|---|------------------------|--|
|    | <b>Atribución</b>      | El beneficiario de la licencia tiene el derecho de copiar, distribuir, exhibir y representar la obra y hacer obras derivadas siempre y cuando reconozca y cite la obra de la forma especificada por el autor o el licenciente. |
|    | <b>No Comercial</b>    | El beneficiario de la licencia tiene el derecho de copiar, distribuir, exhibir y representar la obra y hacer obras derivadas para fines no comerciales.  |
|   | <b>No Derivadas</b>    | El beneficiario de la licencia solamente tiene el derecho de copiar, distribuir, exhibir y representar copias literales de la obra y no tiene el derecho de producir obras derivadas.  |
|  | <b>Compartir Igual</b> | El beneficiario de la licencia tiene el derecho de distribuir obras derivadas bajo una licencia idéntica a la licencia que regula la obra original.  |

El presente trabajo se encuentra bajo la licencia Creative Commons

### Desarrollo del proyecto:

“Sistema Web de gestión de Farmacia, basado en tecnología móvil para el Hospital Básico 7 B.I Loja”

### Licencia:





## **Anexo 12: Anteproyecto**