



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL

CARRERA:

INGENIERÍA DE SISTEMAS

Tesis previa a la obtención del título de:

INGENIERA DE SISTEMAS

TEMA:

“SISTEMA DE RECETA ELECTRÓNICA PARA PRESCRIPCIÓN DE
MEDICAMENTOS”

AUTORA:

PAMELA ALEJANDRA BETANCOURT YÉPEZ

DIRECTORA:

ING. MONICA GÓMEZ

Guayaquil, Marzo del 2015

**DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO
DEL TRABAJO DE GRADO**

Yo, Pamela Alejandra Betancourt Yépez, autorizo a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de grado y su reproducción sin fines de lucro.

Las ideas, conceptos desarrollados, análisis realizados, sistemas creados y las conclusiones del presente trabajo, son de exclusiva responsabilidad de la autora.

Pamela Alejandra Betancourt Yépez
CC.0923770879

DEDICATORIA

Quiero agradecer a Dios que ha sido mi fuente de fortaleza brindando su apoyo para seguir adelante y no dejarme vencer por los obstáculos y por estar conmigo en todo momento y saberme guiar por el buen camino.

A mi querida madre Betty Yépez Rodríguez, que hizo muchos sacrificios para brindarme los estudios y sacarme adelante sola.

A mi querida abuelita Isabel Rodríguez Chila, que me dio su apoyo emocional y por medio de sus oraciones me dio fuerzas para seguir adelante.

A mí querido novio por su apoyo incondicional por estar en los momentos buenos y complicados de mi vida, muchas gracias por su comprensión.

A mis queridos profesores, por la comprensión, dedicación y el empeño de poder compartir sus conocimientos conmigo para hacer de mí una profesional con calidad académica, ética y humana.

A mis amigos y compañeros gracias por todo el tiempo compartido llenos de alegrías y algunas angustias. Por todo lo vivido y aprendido.

Pamela Betancourt

AGRADECIMIENTO

Agradezco principalmente a Dios, gracias a sus bendiciones fue posible realizar este proyecto.

A la **Universidad Politécnica Salesiana sede Guayaquil**, que es la entidad que forma profesionalmente para ser buenos ciudadanos y honrados cristianos, con excelencia academia y humana. Gracias por educarnos y otorgar un Título Académico.

Un agradecimiento especial a la Ing. Mónica Daniela Gómez Ríos, por su gran colaboración, paciencia para atenderme y ayudarme a lo largo del proyecto, siempre predispuesta a colaborar.

Pamela Betancourt

ÍNDICE

CAPITULO 1	3
PLANTEAMIENTO DEL PROBLEMA	3
1.1. Enunciado del problema	3
1.1.1. Factores estructurales	3
1.1.2. Factores intermedios	4
1.1.3. Factores inmediatos	4
1.2. Formulación del Problema.....	4
1.2.1. Objetivo General.....	4
1.2.2. Objetivos Específicos	5
1.3. Justificación	5
1.4. Importancia.....	5
1.5. Necesidad.....	6
1.6. Beneficios	6
1.7. Beneficiarios	7
CAPITULO 2	8
MARCO TEÓRICO.....	8
2.1. Antecedentes Investigativos	8
2.1.1. Aplicaciones web.....	8
2.1.2. Java JDK.....	9
2.1.3. Web Services	10
2.1.4. Java Server Faces JSF	11
2.1.5. Primefaces	11
2.1.6. Hibernate	12
2.1.7. Objetos de Acceso a Datos	13
2.1.8. Composite View (Vistas compuestas).....	14
2.1.9. Model View Controller (Modelo Vista-Controlador) MVC	14
2.1.10. Modelo de persistencia de datos	15
2.2. Fundamentación legal.....	17
CAPITULO 3	18
ANÁLISIS DEL SISTEMA.....	18
3.1. Descripción del Sistema	18
3.2. Requerimientos funcionales	18

3.2.1. Historias.....	18
3.2.2. Cronograma entregable.....	20
3.2.3. Definición de casos de uso	22
3.2.4. Lista de actores	23
3.2.5. Lista de casos de uso	23
3.2.6. Descripción de Casos de Uso	24
3.3. Requerimientos no funcionales	26
3.3.1. Software.....	26
3.3.2. Hardware	27
3.3.3. Definición de Roles en los módulos.....	28
3.4. Modelo de Procesos.....	29
3.4.1. Mapa de Procesos	29
3.4.2. Matriz	30
3.4.3. Procedimiento de Ingreso de Paciente.....	31
3.4.4. Procedimiento Envió de Receta Electrónica	31
3.5. Diagramas de Flujo de Datos.....	32
3.5.1. Diagrama de Flujo del Proceso de Ingreso de Pacientes.....	32
3.5.2. Diagrama de Flujo del Proceso de Ingreso de Antecedentes del Paciente	33
3.5.3. Diagrama de Flujo del Proceso de Ingreso de Consultas Medicas.....	34
3.5.1. Diagrama de Flujo del Proceso de Envió de Receta al Mail del Paciente.	35
3.6. Diagrama Jerárquico del Sistema	36
3.6.1. Diagrama Jerárquico General de la Aplicación.....	36
3.6.2. Diagrama Jerárquico del Ingreso al Sistema	36
3.6.3. Diagrama Jerárquico del Registro del Paciente.....	37
3.6.4. Diagrama Jerárquico del Registro de Antecedentes	37
3.6.5. Diagrama Jerárquico de Consulta Médica.....	38
3.6.6. Diagrama Jerárquico de Prescripción Médica.....	38
CAPITULO 4	39
DISEÑO DEL SISTEMA	39
4.1. Diseño de la Arquitectura del sistema	39
4.1.1. Arquitectura Cliente-Servidor	39
4.1.2. Diseño Arquitectónico	41
4.1.3. Módulos del sistema	41

4.2.	Diagrama de clases del Sistema.....	43
4.2.1.	Detalle Diagrama de clases.....	43
4.3.	Modelo lógico de la base de datos.....	47
4.3.1.	Código Fuente	48
CAPITULO 5		59
IMPLEMENTACIÓN Y PRUEBAS.....		59
5.1.	Capas del Sistema y Comunicación entre Capas.....	60
5.1.1.	Capa de datos.....	61
5.1.2.	Capa de negocio.....	61
5.1.3.	Capa de presentación	61
5.2.	Plan de Pruebas.....	62
5.2.1.	Pruebas unitarias.....	62
5.3.	Resultados de las Pruebas y métricas tomada	66
CAPITULO 6.....		68
CONCLUSIONES Y RECOMENDACIONES.....		68
6.1.	Conclusiones.....	68
6.2.	Recomendaciones	69
6.3.	Bibliografía.....	70
Anexos.....		72

ÍNDICE DE TABLAS

Tabla 1 Historia de usuario de diseño de base de datos	18
Tabla 2 historia de usuario de diagrama de clases	19
Tabla 3 historia de usuario de diseño de receta electrónica	19
Tabla 4 Cronograma entregable	20
Tabla 5 Lista de casos de uso	23
Tabla 6 Caso Uso Ingreso al Portal	24
Tabla 7 Caso uso registro de paciente	24
Tabla 8 Caso Uso Registro Antecedentes	24
Tabla 9 Caso Uso Consulta Médica	25
Tabla 10 Caso Uso Prescripción Médica	25
Tabla 11 Envío de receta médica	26
Tabla 12 Requerimientos no funcionales – Rapidez	26
Tabla 13 Requerimientos no funcionales – Base de datos	26
Tabla 14 Requerimientos no funcionales – Framework.....	27
Tabla 15 Requerimientos no funcionales – IDE	27
Tabla 16 requerimientos no funcionales servidor de aplicaciones.....	27
Tabla 17 Definición de Roles.....	28
Tabla 18 Matriz Pat (Proceso de Consulta y prescripción médica)	30
Tabla 19 Procedimiento Ingreso de Paciente	31
Tabla 20 Procedimiento Envío de Receta Electrónica	31
Tabla 21 Diagrama de Flujo del Proceso de Ingreso de Pacientes.....	32
Tabla 22 Diagrama de Flujo del Proceso de Ingreso de Antecedentes	33
Tabla 23 Diagrama de flujo del proceso de Ingreso de Consultas	34
Tabla 24 Diagrama de Flujo del Proceso de Envío de Receta Médica al Mail.....	35
Tabla 25 Detalles Diagrama de Clases.....	43
Tabla 26 Prueba Unitaria Ingreso al Portal	62
Tabla 27 Prueba Unitaria Ingreso al Portal (Éxito).....	62
Tabla 28 Prueba Unitaria Registro de Paciente.....	63
Tabla 29 Prueba Unitaria Registro de Paciente.....	63
Tabla 30 Prueba Unitaria Consulta Médica	64
Tabla 31 Prueba Prescripción Médica.....	65
Tabla 32 Prueba envió de receta médica.....	65

Tabla 33 Prueba imprimir receta.....	66
Tabla 34 Resultado de Pruebas	67

ÍNDICE FIGURAS

Figura 1 Hibernate Arquitectura	12
Figura 2 Data Access Object.....	14
Figura 3 Aplicación en Java utilizando MVC.....	15
Figura 4 Sentencias SQL Directas	16
Figura 5 Capa de Persistencia Hibernate	16
Figura 6 Casos de uso paciente	22
Figura 7 Caso de uso médico	22
Figura 8 Mapa de Procesos	29
Figura 9 Diagrama Jerárquico General de la Aplicación	36
Figura 10 Diagrama Jerárquico del Ingreso al Sistema	36
Figura 11 Diagrama Jerárquico del Registro del Paciente	37
Figura 12 Diagrama Jerárquico del Registro de Antecedentes	37
Figura 13 Diagrama Jerárquico de Consulta Médico.....	38
Figura 14 Diagrama Jerárquico de Prescripción Medica	38
Figura 15 Arquitectura Cliente-Servidor	40
Figura 16 Diseño Arquitectónico Base de Datos	41
Figura 17 Capas Patrón MVC.....	42
Figura 18 Diagrama de Clases	43
Figura 19 Modelo Lógico BD.....	47
Figura 20 Paquetes	48
Figura 21 Método GETCONECCION.....	49
Figura 22 Método LOGINPROJECT	50
Figura 23 Método CONFIGURE.....	50
Figura 24 Métodos DAO.....	51
Figura 25 RECETADAOIMPLEMENT.....	52
Figura 26 Clases Model	52
Figura 27 Clases Model BD	53
Figura 28 HIBERNATEREVENGE.....	53
Figura 29 Métodos BEAN	54
Figura 30 Método SEND	55
Figura 31 Método imprimir	56
Figura 32 Método DOFILTER	56

Figura 33 Filtro SERVLET	57
Figura 34 Send Message	57
Figura 35 Método de envío de mensaje	58
Figura 36 Método de envío de mensaje 2	58
Figura 37 Esquema de Red	59
Figura 38 Comunicación entre capas del sistema	61
Figura 39 Resultado de pruebas	67

ÍNDICE DE ANEXOS

Anexo A Manual de usuario del Recetario Electrónico.....	72
Anexo B Diccionario de Datos	86
Anexo C Fotos y Certificado	97

RESUMEN

Hoy en día los estilos de vidas tan absorbentes y los avances en medicamentos han desencadenado la automedicación, sin saber que realizarla es muy delicado y perjudicial para la salud. Se debe evitar a toda costa la automedicación, ya que lejos de ayudar, puede perjudicar más el estado de salud.

La automedicación es nociva, ya que se desconoce la dosis y la temporalidad en que se deben consumir ciertos medicamentos y éstos pueden tener efectos secundarios, incluso afectar de manera grave la salud, lo que implica que todo lo expuesto sea un factor principal para la finalidad de este proyecto, que será un sistema de receta electrónica para la prescripción médica.

El sistema permitirá que cualquier persona que acceda a la página web pueda comunicarse con el médico del dispensario UNIMSA en el que se va a implementar este proyecto, mediante el ingreso de los datos y la iteración con el médico el paciente podrá recibir por medio electrónico una prescripción de medicamentos para poder erradicar o tratar las dolencias que lo aquejan, de un modo rápido y gratis.

ABSTRACT

Today's styles so absorbent lives and advances in medicine have triggered self-medication without knowing that realize it is very delicate and unhealthy. Avoid at all costs-medication, as far from help, harm your health.

The automedication is harmful, from the dose and the distribution gear to consume certain medicines and can have side effects is not known still seriously they harm the health, implying that the previous one is a principal factor to the aims this project, which is an electronic prescription for the medical prescription.

The system will allow any person accessing the website to communicate with the medical dispensary UNIMSA in which you will implement this project by entering data and iteration with the doctor the patient will receive by mail one prescription drugs to eradicate or treat ailments that afflict it, a fast and freeway.

INTRODUCCIÓN

En el país existen muchos centros médicos a los cuales las personas acuden por un tratamiento, pero se encuentran con diversos inconvenientes, como es el caso de que tienen que esperar algunos días para ser atendidos, la consulta médica no está al alcance de su bolsillo o simplemente en ocasiones estos dispensarios se encuentran llenos y hace difícil poder consultar a un médico.

El propósito del presente proyecto es el desarrollo de una herramienta tecnológica que pueda cubrir la necesidad de las personas cuando requieran consultar al médico para recibir un tratamiento de una manera fácil, rápida y desde la comodidad del hogar por el momento este servicio será gratuito y no generará costo alguno.

CAPITULO I - EL PROBLEMA, señalará cómo se transfiere desde la idea preliminar hasta el planteamiento del problema mediante un proceso deductivo que visualiza la formulación del problema, con los objetivos de la investigación, la justificación y delimitación correspondiente.

CAPITULO II - EL MARCO TEÓRICO, describe cómo se desarrolla, en qué consiste, los antecedentes investigativos y el contexto legal del proyecto.

CAPITULO III – ANALISIS DEL SISTEMA, describe los procedimientos que se acogieron para el análisis del sistema, como son los requerimientos funcionales y no funcionales del mismo.

CAPITULO IV – DISEÑO DEL SISTEMA, describe los modelos, diagramas y módulos del sistema los cuales van a permitir conocer la arquitectura y el modelamiento de los datos que se necesitaron para diseñar el sistema web.

CAPITULO V – IMPLEMENTACION Y PRUEBAS, presenta la forma en la que se sube el sistema al ambiente de producción describiendo en un diagrama de red y realizando las respectivas pruebas del sistema de prescripción de receta electrónica.

CAPITULO VI – CONCLUSIONES Y RECOMENDACIONES en este capítulo, se plantea las recomendaciones pertinentes para la continuación del proyecto y se finaliza en función de la necesidad de cada persona por obtener un tratamiento médico rápidamente.

CAPITULO 1

PLANTEAMIENTO DEL PROBLEMA

Las personas de la ciudad están obligadas a llamar a la entidad de salud por una simple consulta médica, esto se da por el gran número de personas que desean ser atendidas por un médico, por lo cual se está creando un portal web, el cual permita la interacción de cualquier persona con un médico especializado, éste podrá diagnosticar alguna dolencia que tenga y prescribir una receta, la cual será enviada por correo electrónico, con esto se disminuye el problema de las personas que tengan hasta una simple gripe y tengan que esperar demasiado tiempo para que los atiendan.

1.1. Enunciado del problema

Hoy en día existe una gran demanda de pacientes que necesitan ser atendidos por los diferentes centros médicos de la ciudad, esto ocasiona un grave problema, ya que es difícil atender una gran demanda de personas que acuden por ayuda hospitalaria, además las personas son obligadas a reservar las consultas médicas a fechas futuras lo que puede dar como resultado que cualquier persona agrave su estado de salud.

El sistema de receta electrónica para prescripción de medicamentos, será creado para el dispensario UNIMSA, esto ayudará a bajar la demanda de personas que se hacen atender en el centro de salud, con una innovadora forma de recibir una receta al correo electrónico y poder interactuar con el médico a través de la web por mensajes de texto desde la comunidad de su hogar.

1.1.1. Factores estructurales

Existe la manera tradicional de recetar medicina por parte de los médicos, por medio de un simple papel, en donde se especifica la descripción de los medicamentos, el tiempo y la cantidad de suministrarse, este proceso es tedioso ya que visitar al doctor cada vez que desee una consulta médica sería una pérdida de tiempo, lo que ocasionaría que el paciente pueda ponerse en mal estado, si pasa mucho tiempo hasta el momento de ser atendido. Por este motivo la receta electrónica vendría ser una solución a los problemas de citas médicas.

1.1.2. Factores intermedios

El dispensario UNIMSA está diseñado para brindar una oportuna atención de salud, el cual atiende casos que no requieran de hospitalización, brinda primeros auxilios y realiza prescripciones médicas, en casos que lo amerita. La falta de control de los tiempos utilizados en la consulta y prescripción médica, afecta en la obtención inmediata de un tratamiento médico para el paciente que lo necesita.

1.1.3. Factores inmediatos

Si no se aplica la receta electrónica en el dispensario UNIMSA seguirá la pérdida de tiempo porque no se provee tratamientos inmediatos a cualquier persona que lo requiera y no habrá el uso de la tecnología y seguirá recibiendo en papel la receta prescrita por el médico.

Las consecuencias positivas para el dispensario será la satisfacción que proveerá a cualquier persona que necesite un tratamiento médico o se encuentre enfermo, además el dispensario ganará más pacientes por la manera de atención única en la ciudad.

1.2. Formulación del Problema

- ¿De qué manera se evitaría que las personas reciban un tratamiento médico a futuro y no en el momento en que lo necesitan?
- ¿Qué impacto tendría el uso de una receta electrónica para los pacientes que utilicen este sistema?

1.2.1. Objetivo General

Desarrollar una aplicación web para el dispensario UNIMSA, que permita a cualquier persona recibir una prescripción médica gratuita, a través de la emisión de una receta electrónica, la misma que indicará el tipo de tratamiento u observaciones a seguir.

1.2.2. Objetivos Específicos

- Brindar tratamientos médicos a través de la emisión de la receta electrónica, misma que llegará al correo electrónico del paciente registrado en el portal web de UNIMSA
- Disminuir la alta demanda de pacientes en el centro de salud UNIMSA, al utilizar el servicio desde la comodidad de su hogar sin necesidad de trasladarse al centro de salud
- Fomentar el uso de las tecnologías de información a través del recetario electrónico

1.3. Justificación

Uno de los principales inconvenientes de solicitar una cita médica en cualquier centro o dispensario médico, es el procedimiento que deben seguir los afiliados o pacientes, que consiste en pedir una cita médica por teléfono o solicitarlo por internet para agendar la cita médica para una fecha futura, esto ocasiona que demande mucho tiempo porque si no hay cupo disponible inmediato, tal vez haya en unas dos semanas, por tal motivo usando el sistema de receta electrónica puede ayudar en diversas maneras, ahorrando tiempo a las personas que lo utilicen, es más rápido el tratamiento mediante el sistema de receta electrónica, porque permitirá comunicar por medio de un chat al paciente con el doctor por tal motivo brindará inmediatamente la prescripción médica, siempre y cuando sea de baja complejidad.

Utilizar la receta electrónica permitirá grandes beneficios inmediatos sobre la salud, el cual ayudará un diagnóstico por parte del médico. Actualmente no se cuenta con un sistema que permita al paciente interactuar con el doctor y recibir una prescripción médica desde el hogar.

1.4. Importancia

Modernizar el servicio médico para evitar la manera tradicional de emitir una receta médica, de tal manera que al recibir su receta electrónica sea más fácil visualizar la prescripción médica, debido al tipo de letra que tienen los doctores, a veces es imposible ver con claridad lo que dice en la prescripción. Las personas podrán consultar al médico sobre algún síntoma y poder recibir una receta para

prevenir una enfermedad o prevenir que se agrave, haciendo más fácil el diagnóstico del médico y pueda recetar al paciente en una forma más correcta por el seguimiento del historial del paciente y por integrar la información en tiempo real al historial médico, lo que lleva a establecer uniformidad del procedimiento, cuando mantenga el paciente una consulta online con el médico estará dispuesto a tener una comunicación amable y estará dispuesto a explicarle detalladamente todas las dudas que el paciente tenga como cualquier cita tradicional.

1.5. Necesidad

La mayoría de los hospitales y centros de salud están congestionados: por este motivo se vio la necesidad de crear un portal web para el centro médico UNIMSA con esto ayudará a descongestionar los diferentes centros de atenciones médicas, además de dar soluciones rápidas a enfermedades ambulatorias ya que las enfermedades que existen en su mayoría son ambulatorias y pueden ser tratadas por medio de este sistema. El paciente no visita físicamente al médico si no que obtiene con facilidad su prescripción médica a través del portal web.

1.6. Beneficios

- Obtener una prescripción médica desde la comodidad del hogar
- Ahorro de tiempo al momento de recibir un tratamiento médico
- Acortar distancia, si el paciente vive en un lugar alejado y si se le complica la movilización, el paciente podrá consultar al doctor desde su hogar
- Consultas rápidas, si el doctor le recetó al paciente algún medicamento y le produjo síntomas secundarios podrá informarle inmediatamente al médico y no esperar hasta la próxima consulta para realizar la pregunta
- Tratamiento continuo, si el paciente tiene un tratamiento continuo y tiene que visitar al médico todos los meses y a partir de la segunda visita ya no va al médico por diferentes motivos de complicación. Este sistema ayudará a seguir el tratamiento sin interrupciones
- Acceder a los servicios del centro médico UNIMSA cambiando el modo tradicional de recibir una consulta médica, ya que la persona puede obtener la prescripción médica en forma virtual

1.7. Beneficiarios

Los beneficiarios serán aquellas personas con baja complejidad de prescripción médica, es decir personas a quienes se pueda dar un tratamiento a cualquier enfermedad ambulatoria. Algunas de las enfermedades consideradas como ambulatorias son:

- Cólera
- Diarrea
- Fiebre tifoidea
- Gripe o influenza
- Hepatitis
- Meningitis
- Sarampión
- Otitis
- Varicela

A demás el sistema de receta electrónica será de gran beneficio para las siguientes personas:

- Personas que no puedan movilizarse
- Personas de bajo recursos
- Personas que no estén afiliadas a ningún seguro médico

Cabe recalcar que el uso del sistema es para todos los pacientes que puedan acceder al servicio, el portal web estará orientado a cualquier persona que necesite tener una atención sumamente rápida y accesible.

CAPITULO 2

MARCO TEÓRICO

2.1. Antecedentes Investigativos

Es evidente que la Tecnología de la Información y la Comunicación en general, van a propiciar cambios en la forma de prestar asistencia médica, en donde las innovaciones tecnológicas actúan como soporte de información en el ámbito médico. La receta electrónica es un sistema tecnológico automatizado el cual busca aumentar las actividades profesionales, las cuales realizan la prescripción de medicamentos de manera automatizada.

2.1.1. Aplicaciones web

Es una aplicación que puede ser utilizada para acceder a un servidor Web a través de la Internet o de una Intranet. Son muy populares debido a la facilidad de uso de los navegadores web como clientes "thin". Otra razón importante de su popularidad es la capacidad de actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de potenciales clientes. (Tang, 2012)

Tecnologías

Las tecnologías son un concepto que hace referencia a una amplia variedad de recursos tecnológicos, desarrollados a partir de la informática, que son empleados en las telecomunicaciones. La posibilidad de interactuar a través de redes o en internet con herramientas como la computadora, la telefonía móvil o cualquier otro dispositivo electrónico con capacidad de almacenar, procesar y transmitir información, ha causado una profunda revolución en la manera en que los seres humanos acceden a la información, la generan, difunden y se comunican. (JohnZerzan, 2013)

Servlets

La palabra servlet se deriva de "applet", que utiliza para referirse a pequeños programas escritos en Java que se ejecutan en un navegador Web. En comparación con "servlet" es un programa ejecutado en un servidor Web. El uso más común de

servlets es generar páginas Web dinámicas en base a los parámetros de la solicitud enviada por el navegador Web. (Sánchez, 2012)

JavaScript

JavaScript es un lenguaje interpretado para ser utilizado en páginas Web, y su sintaxis es similar a la del lenguaje Java. Se ha convertido en un lenguaje con todas las funciones de programación que lleva el poder de la computación al navegador para mejorar la usabilidad de aplicaciones Web con técnicas avanzadas, como el Ajax. (Gonzalez, 2013)

Ajax

Según Gabriel González, 2013. Ajax no es una tecnología, es una arquitectura. Es sinónimo de XMLHttpRequest asíncrono. Características:

- Sobre la base de las normas de presentación utilizando XHTML y CSS
- Visualización e interacción dinámica utilizando Document Object Model
- El intercambio de datos y la gestión de uso de XML y XSLT
- Recuperación de datos asincrónica utilizando XMLHttpRequest
- JavaScript para traer estas tecnologías

Ajax mejora la experiencia Web del usuario. Se transforma una experiencia discontinua, donde los usos de esperar a que el servidor responda después de cada solicitud de página, en una experiencia continua y sin fisuras, donde los usuarios interactúan con una interfaz que responde rápidamente, independientemente de las conexiones con el servidor, que son transparentes para el usuario.

2.1.2. Java JDK

Java Development Kit o (JDK), es un software que provee herramientas de desarrollo para la creación de programas en java. (HOLZNER, 2000)

Según Galo Latorre, 2010. Puede instalarse en una computadora local o en una unidad de red. En los sistemas Windows sus variables de entorno son:

- JAVAPATH: es un path completo del directorio donde está instalado el JDK.
- CLASSPATH: son las librerías o clases de usuario
- PATH: variable donde se agrega la ubicación de JDK

Los programas más importantes que se incluyen son:

- Appletviewer: es un visor de applet para generar sus vistas previas, ya que un applet carece de método main y no se puede ejecutar con el programa java.
- Javac: es el compilador de JAVA
- Java: es el intérprete de JAVA
- Javadoc: genera la documentación de las clases java de un programa

El JDK tiene el compilador que permite convertir código fuente en bytecode, es decir, el código "maquina" de la máquina virtual de Java.

JVM

La JVM es el programa que interpreta el código Java mientras que las librerías de clases estándar son las que implementan el API de Java. Ambas JVM y API deben ser consistentes entre sí, de ahí que sean distribuidas de modo conjunto.

Un usuario sólo necesita el JRE para ejecutar las aplicaciones desarrolladas en lenguaje Java, mientras que para desarrollar nuevas aplicaciones en dicho lenguaje es necesario un entorno de desarrollo, denominado JDK, que además del JRE (mínimo imprescindible) incluye, entre otros, un compilador para Java. (Galo, jvm, 2010)

2.1.3. Web Services

Los Web Services (Servicios Web) son una estructura estándar basado en un software, que acepta solicitudes con un formato especial de otras entidades de software en máquinas remotas a través de protocolos de transporte de comunicaciones neutras, produciendo aplicaciones de respuestas específicas. En resumen los web services tienen estas características: (Servicios-Web, 2013)

- Basado en estándares
- Idioma agnóstico
- Solicitudes con formato
- Máquinas remotas
- Proveedores Neutral de Transporte neutral
- Aplicación específica Respuestas

2.1.4. Java Server Faces JSF

Según Gabriel González, 2013. Java Server Faces (JSF) es un framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE. JSF usa Java Server Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas, pero también se puede acomodar a otras tecnologías como XUL.

Acerca de JSF, deben tenerse en cuenta dos consideraciones importantes:

- Se trata de una tecnología que ejecuta del lado del servidor y no del lado del cliente.
- La interfaz de usuario es tratada como un conjunto de componentes UI. Este es un concepto fundamental para la comprensión de la tecnología JSF.

JSF incluye:

- Según Gabriel González, 2013. Un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar eventos, validar entrada, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad
- Un conjunto por defecto de componentes para la interfaz de usuario
- Dos librerías de etiquetas personalizadas para JavaServer Pages que permiten expresar una interfaz JavaServer Faces dentro de una página JSP
- Un modelo de eventos en el lado del servidor
- Beans administrados.

2.1.5. Primefaces

Según Chauhaan Nagesh 2013. PrimeFaces es una librería de componentes visuales open source desarrollada y mantenida por Prime Technology, una compañía Turca de IT especializada en consultoría ágil, JSF, Java EE y Outsourcing. El punto fuerte de **PrimeFaces** es la sencillez de instalación y lo poco pesado que es. El mantenerlo liviano, sin complicaciones a la hora de instalarlo, es decir, sin dependencias ni configuraciones, hace que se pueda usar en unos pocos segundos.

Las principales características de Primefaces son:

- Soporte nativo de Ajax, incluyendo Push/Comet

- Kit para crear aplicaciones web para móviles
- Es compatible con otras librerías de componentes, como JBoss RichFaces
- Uso de javascript no intrusivo (no aparece en línea dentro de los elementos, sino dentro de un bloque <script>)
- es un proyecto open source, activo y bastante estable entre versiones

2.1.6. Hibernate

Arquitectura de Hibernate

Hibernate consiste en una arquitectura de tres capas como se muestra en la siguiente imagen:

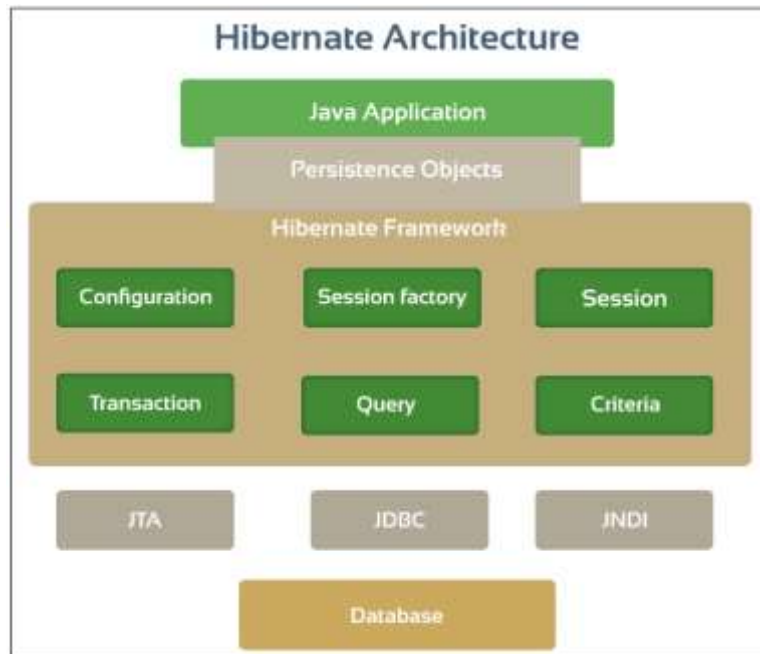


Figura 1 Hibérnate Arquitectura

Elaborado por: Oodles Technologies Ltd. (MANEESH & SONIA, 2009)

Capa de aplicaciones Java:

Con el fin de conservar los datos a la base de datos de una instancia de clase de entidad que se crea este objeto se llama objeto transitorio ya que aún no están asociados con la sesión o todavía no persistieron a una base de datos. (Nagesh, 2013)

Hibernate Framework Capa:

Para guardar el objeto en la base de datos, se crea una instancia de la interfaz SessionFactory. Cargas sessionFactory Hibernate archivo de configuración (hibernate.cfg.xml) y con la ayuda de TransactionFactory y ConnectionProvider implementa todos los ajustes de configuración en una base de datos. Cada conexión de base de datos en Hibernate se crea mediante la creación de una instancia de la interfaz Session. Sesión representa una única conexión con la base de datos. Objetos de sesión son creados a partir de objetos SessionFactory. (Nagesh, 2013)

La capa de base de datos:

La mayoría de las veces tratan con una base de datos relacional. Los datos se representan en forma de tabla y las filas. (Nagesh, 2013)

2.1.7. Objetos de Acceso a Datos

Este patrón de diseño propone separar y encapsular toda la lógica de acceso a los datos que maneja la aplicación de la implementación, logrando de esta manera desligar la lógica de negocio y presentación del mecanismo de manejo del repositorio de datos que se utilice para almacenar la información. Cuando la capa de lógica de negocio necesite interactuar con la base de datos, va a hacerlo a través de la API que le ofrece DAO. Generalmente esta API consiste en métodos CRUD (Create (Crear), Read (Leer), Update (Actualizar) y Delete (Eliminar)). (Mesa & Torres, 2012)

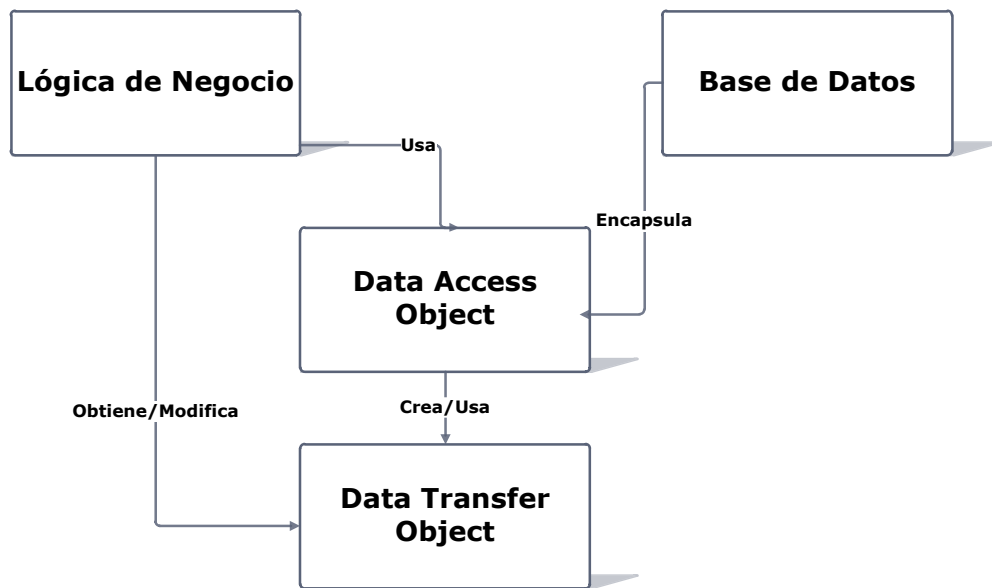


Figura 2 Data Access Object

Elaborado por: Autora

2.1.8. Composite View (Vistas compuestas)

Este patrón proporciona un mecanismo para combinar modularmente, las porciones atómicas de una vista completa, las páginas son construidas uniendo código en el formato dentro de cada vista. Las JSP, Java Server Pages, y los servlets proporcionan mecanismos sencillos para poder incluir porciones de una página en otra (de modo estático o dinámico). Este esquema facilita el mantenimiento de las páginas. (Vera Santana & Lara Vásquez, 2014)

2.1.9. Model View Controller (Modelo Vista-Controlador) MVC

Model View Controller (MVC) es uno de los modelos más recomendados para el diseño de aplicaciones interactivas. MVC separa detalles del diseño (persistencia, presentación, y control de los datos), disminuye la duplicación de código, centraliza el control de la aplicación y hace que los cambios o actualizaciones sean más fácilmente manejables. Un diseño MVC puede centralizar el control de funcionalidades como el uso de seguridad, de sesión, y el flujo de la pantalla. MVC puede adaptarse a nuevas fuentes de datos, creando código que adapta la nueva fuente con las pantallas. (Vera Santana & Lara Vásquez, 2014)

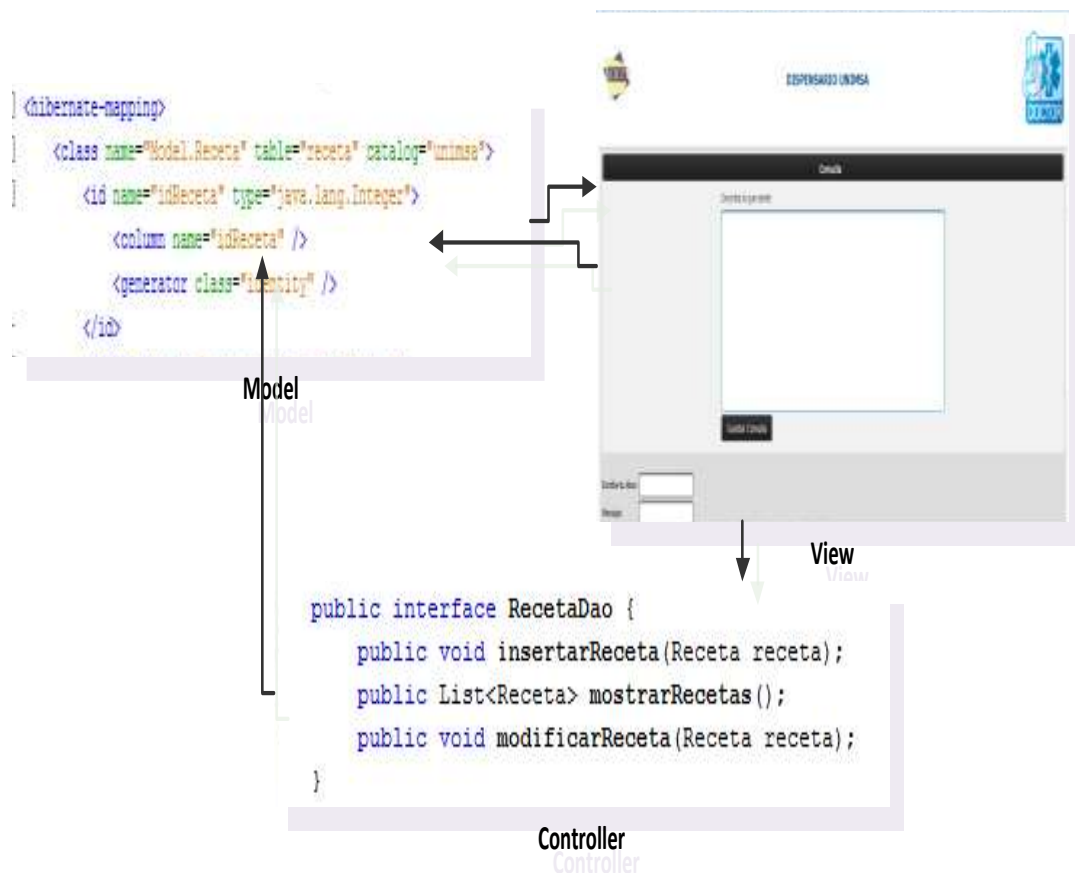


Figura 3 Aplicación en Java utilizando MVC

Elaborado por: Autora

2.1.10. Modelo de persistencia de datos

El modelo utilizado para la persistencia de datos es Hibérnate. Hibérnate es un poderoso objeto de alto rendimiento / persistencia relacional y servicio de consulta. Hibérnate permite desarrollar clases persistentes siguiendo el lenguaje orientado a objetos - incluyendo la asociación, herencia, polimorfismo, la composición y las colecciones. Hibérnate permite expresar consultas en su propia extensión de SQL portátil (HQL), así como en SQL nativo, o con un API de ejemplo o criterio orientados a objetos. (Vera Santana & Lara Vásquez, 2014)

En las siguientes imágenes se muestra como comparar el modo de conexión usado por estas 2 opciones y la diferencia entre los mismos:

En la figura 4 se observa cómo accede a los datos mediante sentencias SQL las cuales son ejecutadas desde las clases de negocios ya que contienen toda la lógica de manipulación de los datos.

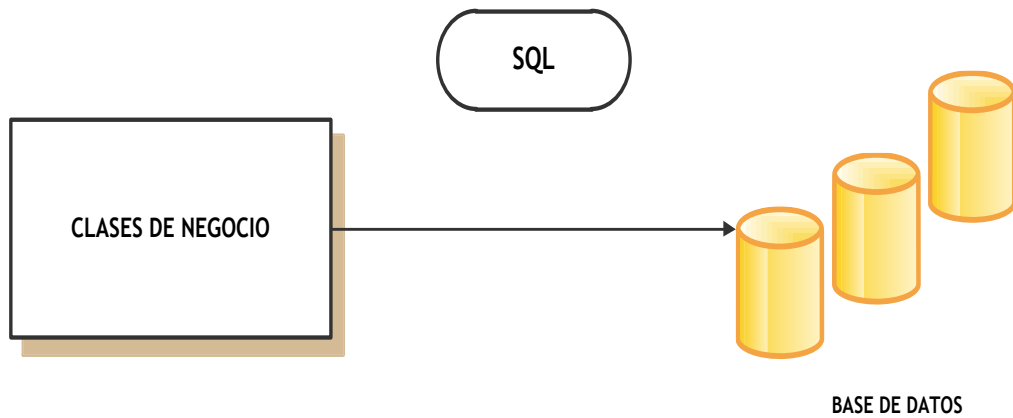


Figura 4 Sentencias SQL Directas

Elaborado por: Autora

En la figura 5 se observa que se accede a los datos mediante sentencias SQL a través del objeto hibernate el cual permite expresar consultas desde su propio entorno con el fin de manipular los datos de la base.

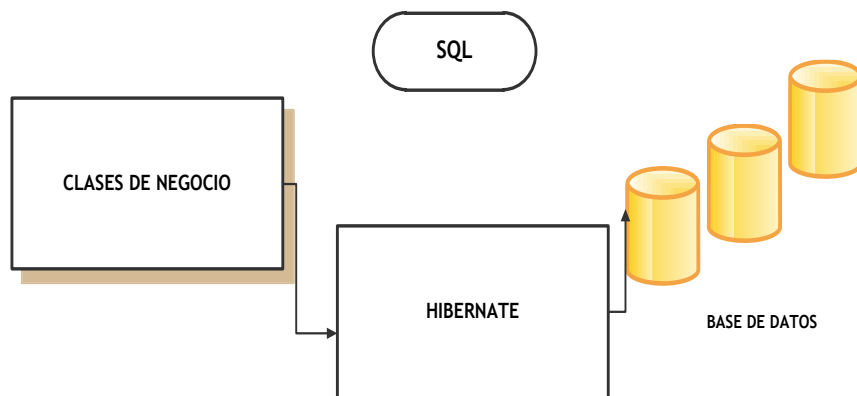


Figura 5 Capa de Persistencia Hibernate

Elaborado por: Autora

2.2. Fundamentación legal

Se debe tener en cuenta que el portal web para el dispensario médico UNIMSA, tendrá un único administrador que será el doctor dueño del dispensario, este mismo tendrá la potestad de administrar el sistema, además podrá decidir si en un futuro cobrará este servicio o lo dejara funcionando gratuitamente. La prescripción médica que se envié a cualquier persona estará con la firma escaneada del mismo doctor, con su respectivo sello, número de registro sanitario y logo que lo diferencia de cualquiera de los demás centros de salud.

El médico del dispensario UNIMSA en donde se aplicará el recetario electrónico por medio de un documento que se adjunta en los anexos, certifica que el desarrollo realizado en este proyecto cumple con las necesidades o especificaciones que el solicito al principio del desarrollo de esta aplicación web, con esto queda constancia que el proyecto de recetario electrónico está totalmente legalizado.

CAPITULO 3

ANÁLISIS DEL SISTEMA

3.1. Descripción del Sistema

Es un sistema que permite el manejo de la información de cualquier persona que necesiten un tratamiento médico, esta información podrá ser consultada por el médico del dispensario el cual tendrá acceso al sistema como administrador y podrá ver las consultas que han enviado los pacientes, luego de esto y por toda la información chequeada el podrá determinar o prescribir un tratamiento médico que será enviado al mail del paciente y podrá ver su tratamiento médico.

3.2. Requerimientos funcionales

3.2.1. Historias

Tabla 1 Historia de usuario de diseño de base de datos

Historias de usuario	
Número: 1	Nombre de usuario:
Nombre de historia: Diseño base de datos	
Programadores responsables:	Pamela Betancourt Yépez
Descripción: Lo que se debe realizar es el diseño del modelo Entidad - Relación de la base de datos que soportará el proyecto.	
Observaciones: Se analizará la problemática y en base a eso se realizará el esquema de datos.	

Elaborado por: Autora

Tabla 2 historia de usuario de diagrama de clases

Historias de usuario	
Número: 2	Nombre de usuario:
Nombre de historia: Diagrama de clases del sistema	
Programadores responsables:	Pamela Betancourt Yépez
Descripción: Lo que se debe realizar es el diagrama de clases que contendrá el sistema.	
Observaciones: Se analizará la problemática y en base a eso se realizará el diagrama de clases del sistema.	

Elaborado por: Autora

Tabla 3 historia de usuario de diseño de receta electrónica

Historias de usuario	
Número: 3	Nombre de usuario:
Nombre de historia: Diseño de receta electrónica	
Programadores responsables:	Pamela Betancourt Yépez
Descripción: Lo que se debe realizar es el diseño de la receta electrónica, la cual incluye las prescripciones e indicaciones médicas, los datos del paciente, el sello y registro sanitario del dispensario médico UNIMSA	
Observaciones: Se analizará la problemática y en base a eso se realizará el diseño de la receta que recibirá el paciente.	

Elaborado por: Autora

3.2.2. Cronograma entregable

Tabla 4 Cronograma entregable

N°	Actividad	Fecha de Inicio	Fecha de Terminación	Actividad predecesora
1	Sistema de Receta Electrónica para Prescripción de Medicamentos	28/08/2014	27/09/2014	
2	Determinación de los Requerimientos del Sistema	28/08/2014	30/08/2014	
3	Unión de información necesaria para el proyecto.	28/08/2014	02/09/2014	
4	Elaboración de lista Requerimientos de la aplicación web.	02/09/2014	03/09/2014	3
5	Análisis de los requerimientos.	04/09/2014	06/09/2014	4
6	Propuesta de desarrollo de la aplicación web	07/09/2014	07/09/2014	5
7	Diseño del Sistema	08/09/2014	09/10/2014	6
8	Establecer de estándares de diseño en las pantallas de la aplicación web.	10/09/2014	10/09/2014	7
9	Creación del diseño de la aplicación web de acuerdo a los requerimientos de sistema	10/09/2014	15/09/2014	8
10	Creación de la base de datos UNIMSA	16/09/2014	19/09/2014	8
11	Modelo entidad relación de la base de datos UNIMSA	19/09/2014	19/09/2014	10
12	Desarrollo del Sistema	06/10/2014	08/10/2014	11
13	Establecer de estándares de desarrollo de la aplicación web.	08/10/2014	08/10/2014	12
14	Creación de las tablas de la base de	20/10/2014	25/10/2014	13

	datos UNIMSA			
15	Modulo: Registro de Paciente	26/10/2014	25/11/2014	12
16	Modulo: Consulta Médica	26/11/2014	25/12/2014	12
17	Modulo: Prescripción de receta medica	26/12/2014	26/01/2014	12
18	Modulo: Envío de Receta médica al paciente	26/01/2015	26/02/2015	12
19	Documentación	26/02/2015	26/02/2015	12
20	Creación manual de usuario y diagrama de clases de la aplicación web.	27/02/2015	05/03/2015	19
21	Creación del diccionario de datos y diagrama entidad relación con los estándares de base de datos de la aplicación web.	06/03/2015	08/03/2015	20
22	Pruebas del Sistema	09/03/2015	10/03/2015	21
23	Pruebas de la Integración de todos los módulos.	11/03/2015	12/03/2015	22
24	Evaluación	12/03/2015	13/03/2015	23
25	Evaluación y retroalimentación.	14/03/2015	15/03/2015	24
26	Corrección de la evaluación y errores.	16/03/2015	17/03/2015	25
27	Implementación	18/03/2015	19/03/2015	26
28	Implementación en el servidor web de internet.	20/03/2015	21/03/2015	26
29	Presentación del proyecto.	30/03/2015	30/03/2015	26
30	Entrega de informes.	30/03/2015	30/03/2015	29

Elaborado por: Autora

3.2.3. Definición de casos de uso

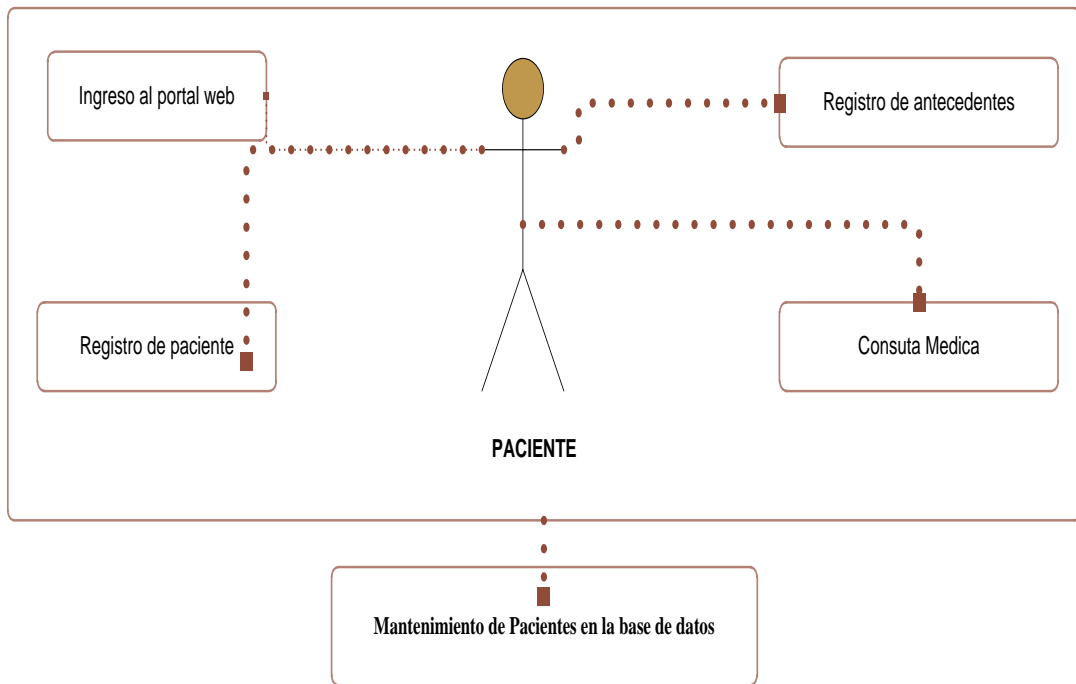


Figura 6 Casos de uso paciente

Elaborado por: Autora

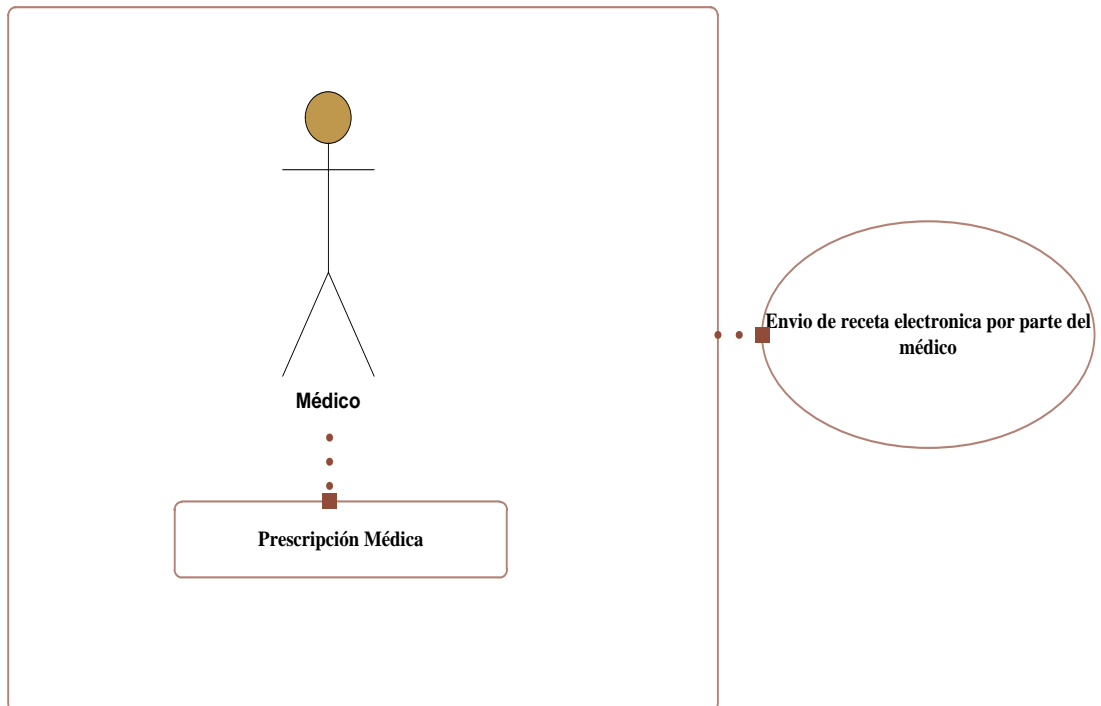


Figura 7 Caso de uso médico

Elaborado por: Autora

3.2.4. Lista de actores

El sistema web contará con dos actores los cuales son sumamente importantes para que el flujo de la aplicación cumpla sus objetivos.

Nombre: Paciente

Descripción: Persona que desea obtener un tratamiento médico

Tipo: Primario

Nombre: Médico

Descripción: Persona que elabora y envía la receta médica

Tipo: Primario

3.2.5. Lista de casos de uso

Tabla 5 Lista de casos de uso

IDENTIFICACION CASO DE USO	NOMBRES CASO DE USO
1	Ingreso al Portal
2	Registro Paciente
3	Registro Antecedentes
4	Consulta Médica
5	Prescripción Medica
6	Envío de receta médica

Elaborado por: Autora

3.2.6. Descripción de Casos de Uso

Tabla 6 Caso Uso Ingreso al Portal

CU:1	Ingreso al Portal
Descripción:	Procedimiento para ingresar al portal web
Observaciones:	Deberá registrarse previamente
Escenarios:	
Ingreso del usuario y la clave (acceso denegado)	
Ingreso del usuario y la clave (bienvenido al portal)	

Elaborado por: Autora

Tabla 7 Caso uso registro de paciente

CU:2	Registro Paciente
Descripción:	Procedimiento para registrar los datos en el portal web
Observaciones:	Formulario de registro de paciente
Escenarios:	
Ingreso de datos personales	
Ingreso de usuario y clave	

Elaborado por: Autora

Tabla 8 Caso Uso Registro Antecedentes

CU:3	Registro Antecedentes
Descripción:	Procedimiento para ingresar registrar los antecedentes del paciente
Observaciones:	Ingresar la información que se crea conveniente
Escenarios:	
Ingresar los antecedentes del paciente	

Elaborado por: Autora

Tabla 9 Caso Uso Consulta Médica

CU:4	Consulta Médica
Descripción:	Procedimiento para que el paciente haga una consulta al médico
Observaciones:	Página para interactuar con el médico
Escenarios:	
	Ingreso de los síntomas que le aquejan al paciente
	Chat interno en el portal web

Elaborado por: Autora

Tabla 10 Caso Uso Prescripción Médica

CU:5	Prescripción Médico
Descripción:	El médico genera el tratamiento
Observaciones:	El médico da una vista a la información del paciente
Escenarios:	
	El médico verifica la información del paciente
	El médico ingresa el tratamiento
	El médico imprime la receta

Elaborado por: Autora

Tabla 11 Envío de receta médica

CU:6	Envío de receta médica
Descripción:	El médico envía la receta al email del paciente
Observaciones:	La receta contiene el tratamiento médico que ayudara al paciente con sus dolencias
Escenarios:	
El médico verifica la información del paciente	
El médico ingresa el tratamiento	
El médico envía receta	
El médico imprime la receta	

Elaborado por: Autora

3.3. Requerimientos no funcionales

3.3.1. Software

Tabla 12 Requerimientos no funcionales – Rapidez

ID:	Rapidez	Relación:	
Descripción: Tiempo de respuesta de la página web			
La aplicación web en general no demorará más de 10 segundos en cargar su contenido. Eso demostraría el tiempo de respuesta óptimo que posee.			

Elaborado por: Autora

Tabla 13 Requerimientos no funcionales – Base de datos

ID:	Base de datos	Relación:	
Descripción: Motor de base de datos Mysql			
La versión del gestor de base de datos que se utiliza es la 6.2, la cual permitirá realizar las diferentes transacciones en el portal web			

Elaborado por: Autora

Tabla 14 Requerimientos no funcionales – Framework

ID:	Framework	Relación:	
Descripción: Framework de desarrollo			
Para la creación de las páginas web se utilizarán dos Framework de desarrollo como son Primefaces e Hibérnate, los cuales permitirán un mejor diseño de las vistas del portal web para esto se utilizará las versiones 4.0 y 2.0			

Elaborado por: Autora

Tabla 15 Requerimientos no funcionales – IDE

ID:	IDE	Relación:	
Descripción: IDE de desarrollo			
Para el desarrollo de la aplicación web se utilizará el netbeans IDE versión 7.4			

Elaborado por: Autora

3.3.2. Hardware

Tabla 16 requerimientos no funcionales servidor de aplicaciones

ID:	Servidor Web	Relación:	
Descripción: Servidor de aplicaciones web			
La aplicación se subirá sobre un servidor de aplicaciones Glassfish 4.0, el equipo contiene un procesador core 2 duo con 2 GB de memoria y sistema operativo Linux			

Elaborado por: Autora

3.3.3. Definición de Roles en los módulos

Tabla 17 Definición de Roles

Usuario	Rol	Módulo	Observación
Médico	Admin	Módulo de prescripción medica	Permite al médico prescribir una receta de acuerdo a lo que ingreso el paciente y lo que le aqueja, además podrá enviar el correo electrónico al paciente con la receta médica.
Paciente	User	Módulo de registro del paciente	Para acceder a los servicios del portal web el paciente necesitara ingresar sus datos a través del formulario de registro del paciente.
		Módulo de registro de antecedentes del paciente	El paciente deberá llenar el formulario de antecedentes en donde va especificar información necesaria para que el médico pueda evaluar su estado de salud.
		Módulo de consulta medica	En este módulo el paciente podrá chatear con el doctor y podrá ingresar los síntomas que presenta para que luego el doctor pueda prescribir la receta.

Fuente: Autora

3.4. Modelo de Procesos

3.4.1. Mapa de Procesos

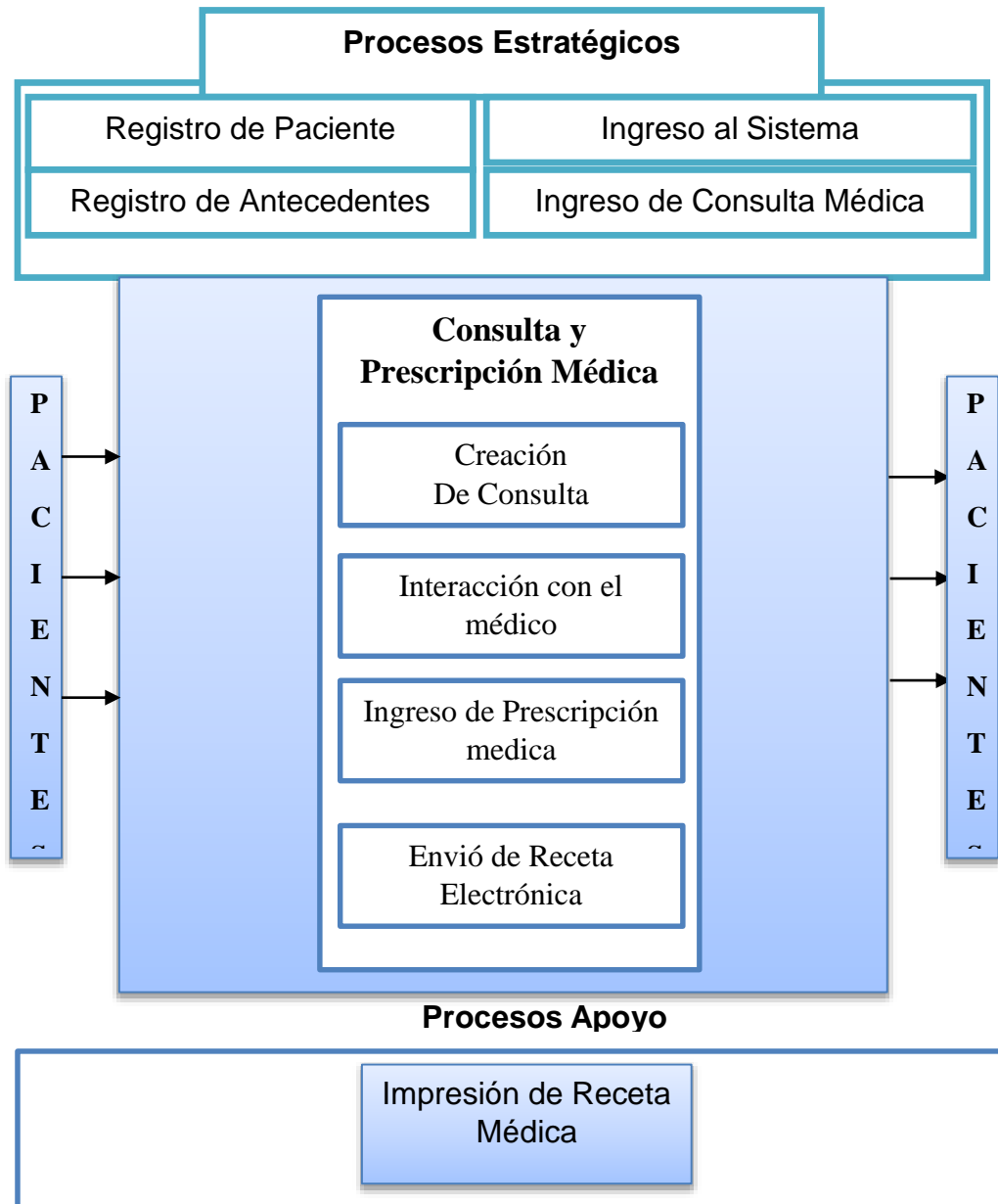


Figura 8 Mapa de Procesos

Elaborado por: Autora

3.4.2. Matriz

Tabla 18 Matriz Pat (Proceso de Consulta y prescripción médica)

MACROPROCESO	PROCESOS	ACTIVIDADES	TAREAS
ADMINISTRACIÓN Y GESTIÓN DE LOS PROCESOS DEL RECETARIO ELECTRONICO	CONSULTA Y PRESCRIPCION MEDICA	Creación de Consulta	Ingreso de consulta para el médico
			Grabar consulta medica
		Interacción con el médico	Chat interactivo entre el médico y los pacientes
		Ingreso de Prescripción Medica	Consultar pacientes que no se le ha prescrito una las indicaciones medicas
			Ver el estado del paciente
Envió de Receta Electrónica	Enviar la receta electrónica al paciente		

Elaborado por: Autora

3.4.3. Procedimiento de Ingreso de Paciente

Tabla 19 Procedimiento Ingreso de Paciente

Actividades	Responsables
El paciente tendrá que registrar sus datos además de ingresar usuario y password para poder acceder al sistema. El paciente podrá acceder mediante su usuario y contraseña al portal web. El paciente solo podrá ver la pantalla de consulta médica y si no ha ingresado sus antecedentes deberá registrarlos.	Paciente

Elaborado por: Autora

3.4.4. Procedimiento Envío de Receta Electrónica

Tabla 20 Procedimiento Envío de Receta Electrónica

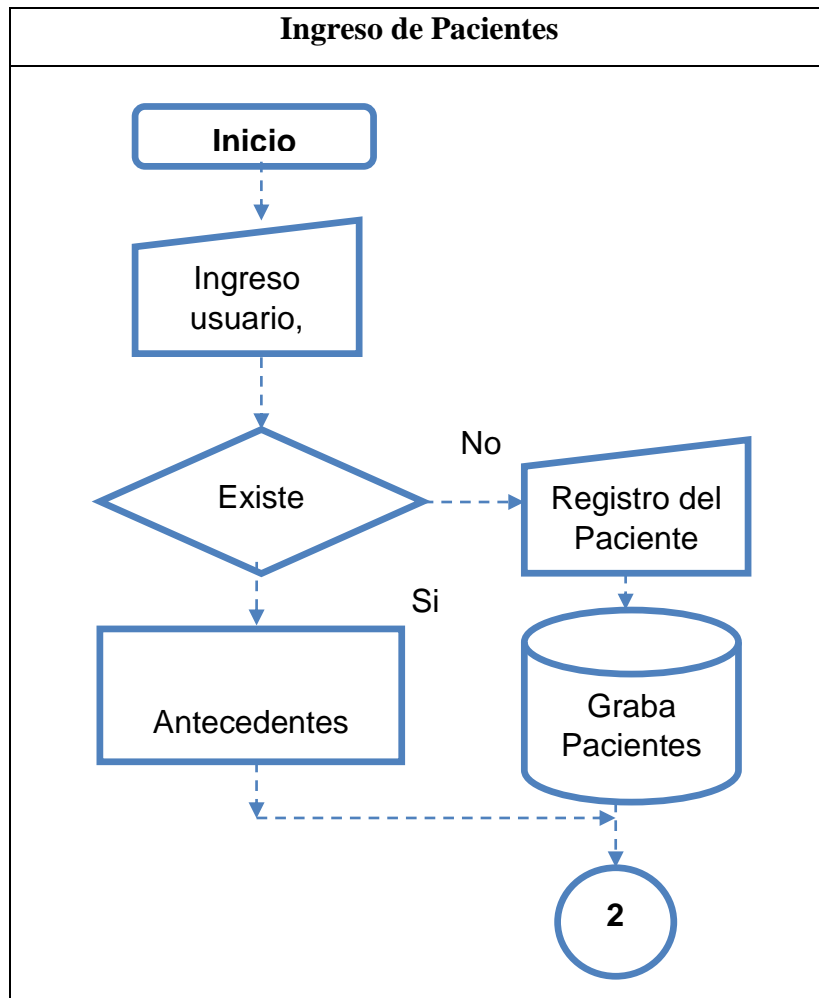
Actividades	Responsables
El médico ingresa al sistema con su usuario y clave respectiva, consulta a los pacientes que no han recibido tratamiento. El médico prescribe el tratamiento del paciente que lo necesite, consultando lo que le aqueja y sus antecedentes. El médico envía la receta electrónica al correo del paciente.	Médico

Elaborado por: Autora

3.5. Diagramas de Flujo de Datos

3.5.1. Diagrama de Flujo del Proceso de Ingreso de Pacientes

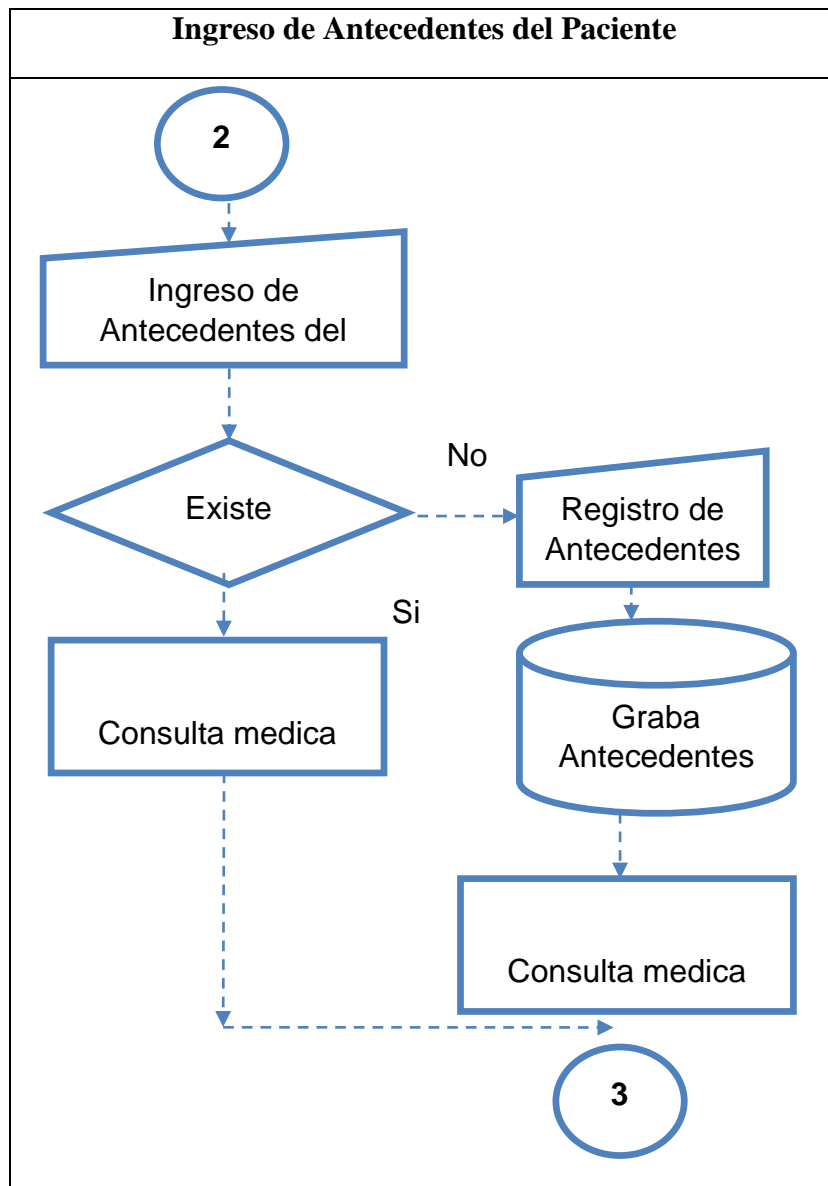
Tabla 21 Diagrama de Flujo del Proceso de Ingreso de Pacientes



Elaborado por: Autora

3.5.2. Diagrama de Flujo del Proceso de Ingreso de Antecedentes del Paciente

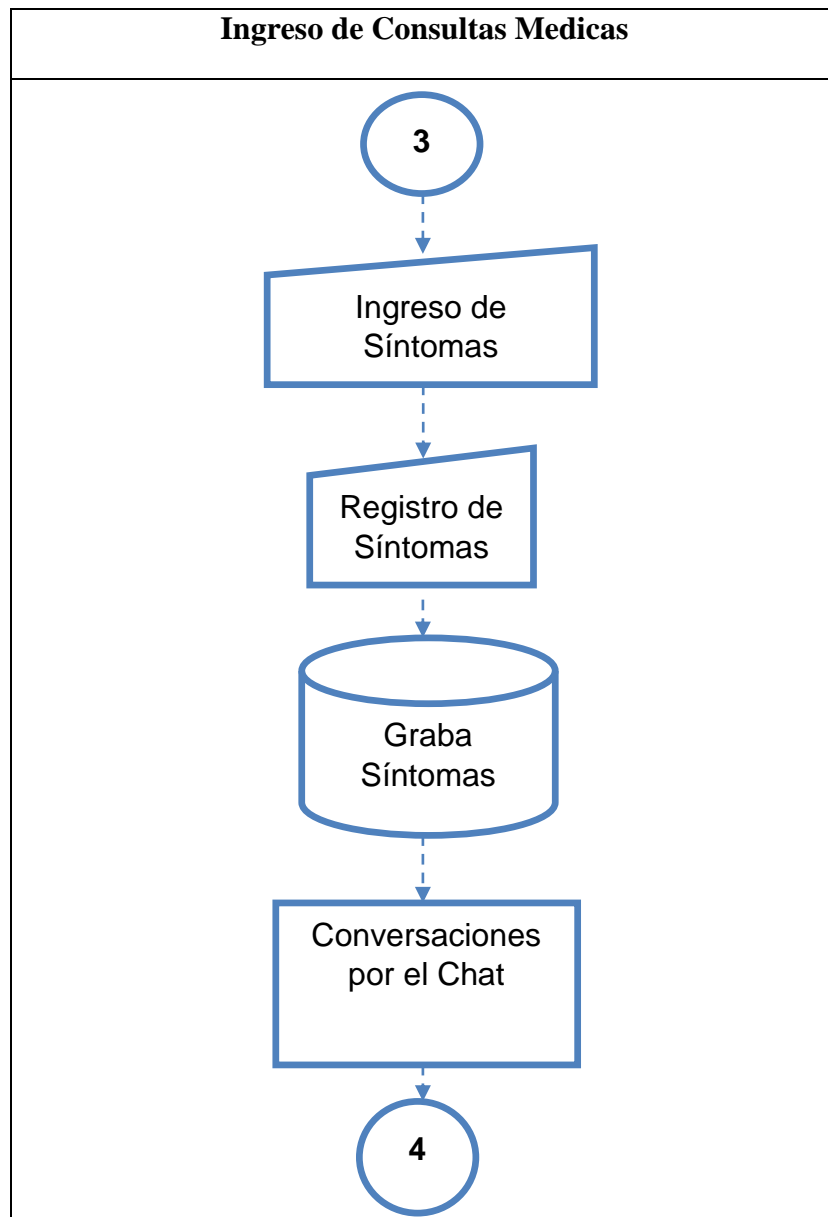
Tabla 22 Diagrama de Flujo del Proceso de Ingreso de Antecedentes



Elaborado por: Autora

3.5.3. Diagrama de Flujo del Proceso de Ingreso de Consultas Medicas

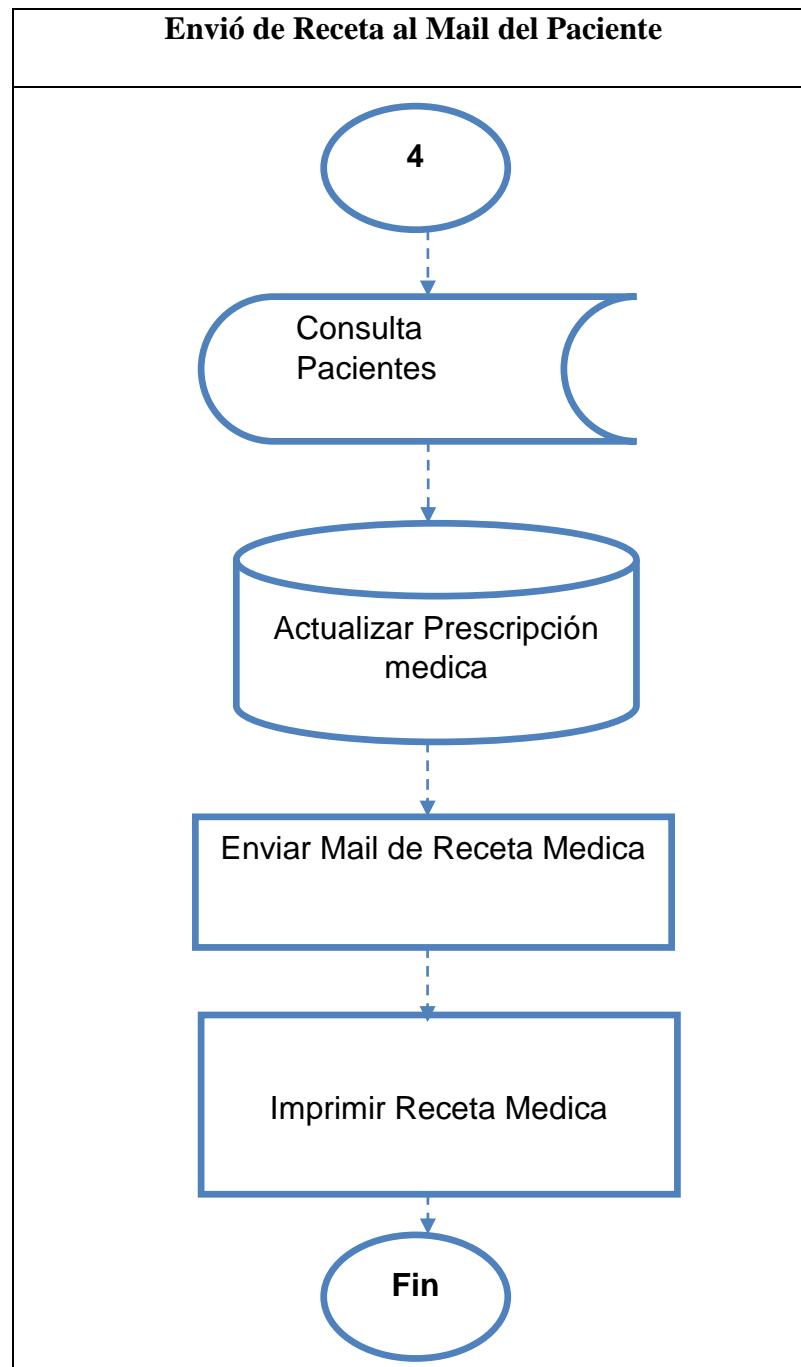
Tabla 23 Diagrama de flujo del proceso de Ingreso de Consultas



Elaborado por: Autora

3.5.1. Diagrama de Flujo del Proceso de Envío de Receta al Mail del Paciente

Tabla 24 Diagrama de Flujo del Proceso de Envío de Receta Médica al Mail



Elaborado por: Autora

3.6. Diagrama Jerárquico del Sistema

3.6.1. Diagrama Jerárquico General de la Aplicación

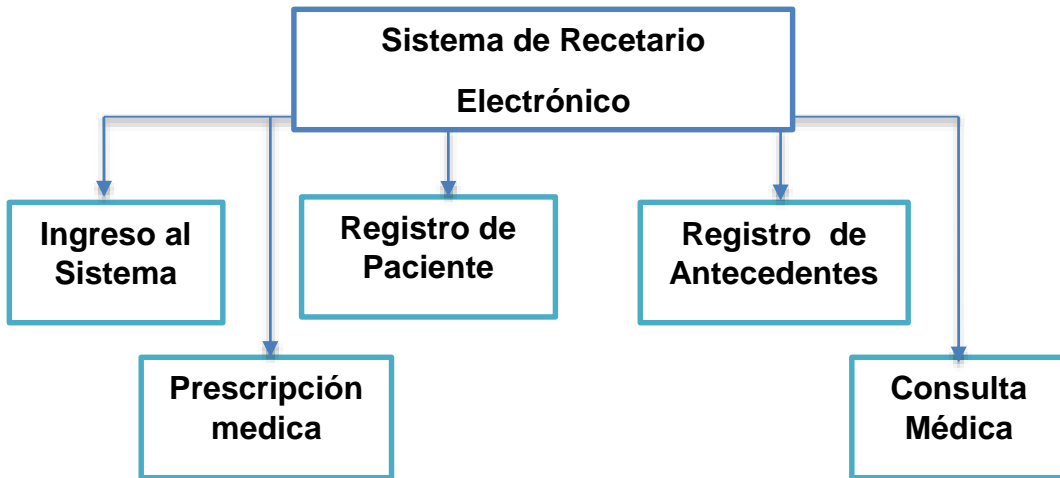


Figura 9 Diagrama Jerárquico General de la Aplicación

Elaborado por: Autora



Figura 10 Diagrama Jerárquico del Ingreso al Sistema

Elaborado por: Autora

3.6.3. Diagrama Jerárquico del Registro del Paciente

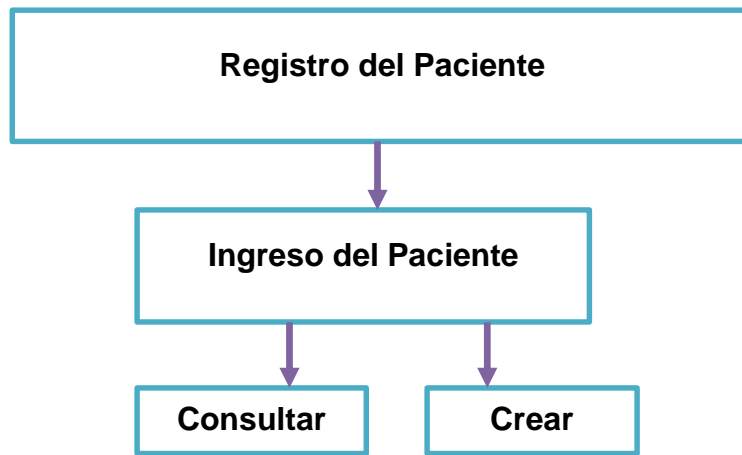


Figura 11 Diagrama Jerárquico del Registro del Paciente

Elaborado por: Autora

3.6.4. Diagrama Jerárquico del Registro de Antecedentes

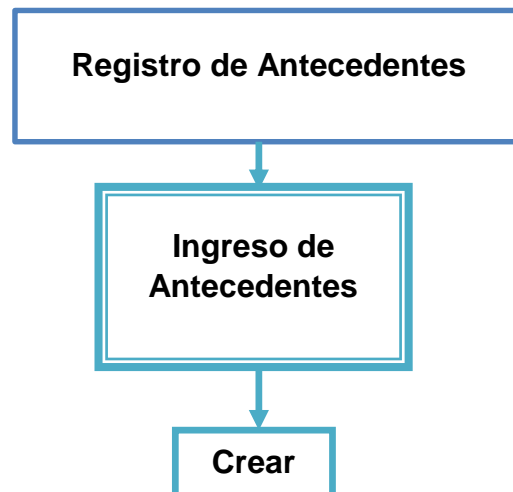
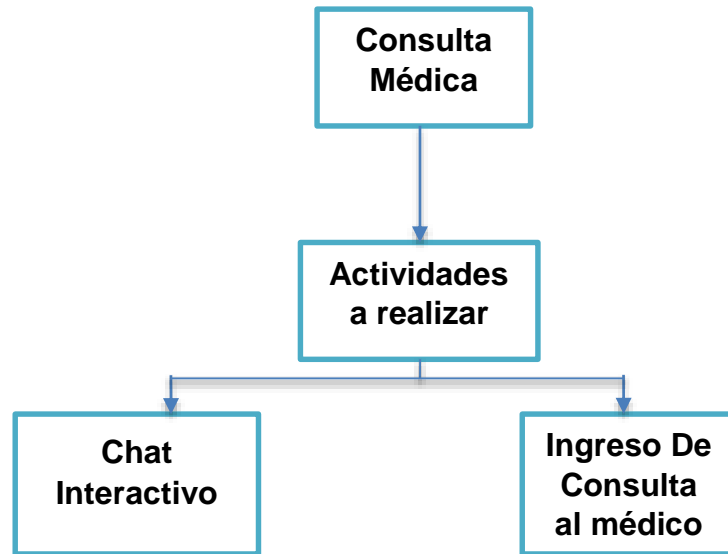


Figura 12 Diagrama Jerárquico del Registro de Antecedentes

Elaborado por: Autora

3.6.5. Diagrama Jerárquico de Consulta Médica



3.6.6. Diagrama Jerárquico de Prescripción Médica

Figura 13 Diagrama Jerárquico de Consulta Médico

Elaborado por: Autora

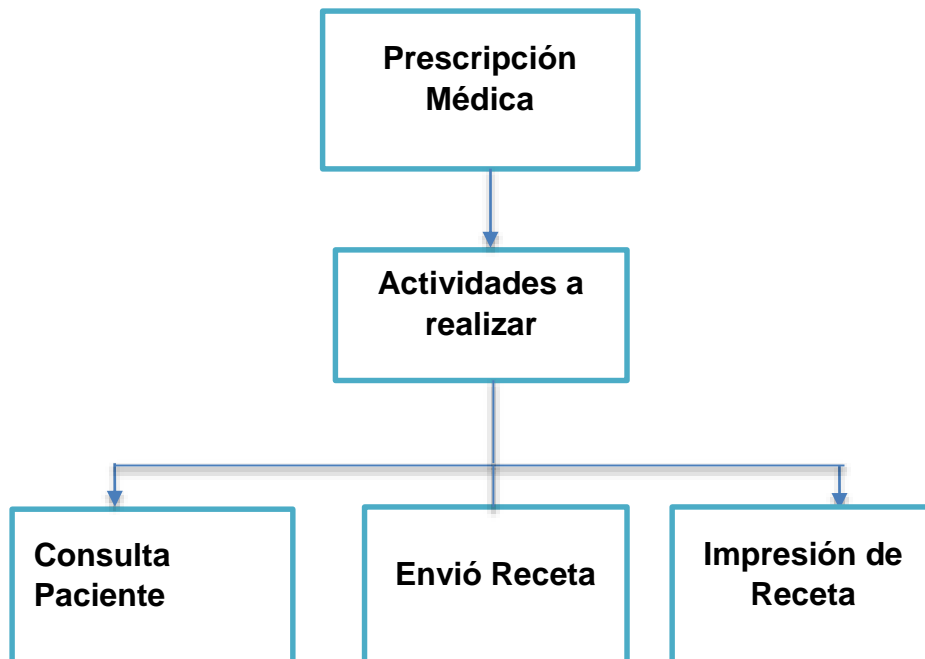


Figura 14 Diagrama Jerárquico de Prescripción Médica

Elaborado por: Autora

CAPITULO 4

DISEÑO DEL SISTEMA

4.1. Diseño de la Arquitectura del sistema

4.1.1. Arquitectura Cliente-Servidor

Se utilizará la arquitectura cliente-servidor en donde los pacientes son una estación de trabajo que reciben los servicios que ofrece un servidor (ejemplo: un servidor web). El servidor ejecuta el software y se ocupa de las funciones de la lógica de negocio de nuestro portal, cuando se realice un cambio en el Servidor automáticamente se podrá ver registrado dicha modificación del lado del cliente, manteniendo la integridad de la información.

Generalmente el lado del paciente tendrá las diferentes validaciones y se guardará la información necesaria para que luego el médico pueda obtener los datos que le van a ayudar a prescribir los medicamentos, en el lado del servidor es donde se encuentra la lógica del negocio, los datos y archivos que puede almacenar el sistema de receta electrónica. El objetivo de la página web a desarrollar es que el paciente pueda registrarse con la condición de que tenga una ficha médica activa, para luego de esto pueda interactuar con el doctor por medio de un chat en un horario determinado, para lo anterior se recurrirá a una arquitectura Cliente-Servidor, ya que el paciente estará interactuando con el doctor. El doctor podrá determinar si el paciente necesita una prescripción médica inmediata, la cual le hará llegar el doctor a su correo electrónico o una consulta física.

En la figura 15 se observa que el paciente realiza una petición http accediendo al portal web, el cual se encuentra sobre un servidor web glassfish 4 que tiene como función ofrecer los servicios necesarios para ejecutar todo el flujo del negocio de este proyecto web.

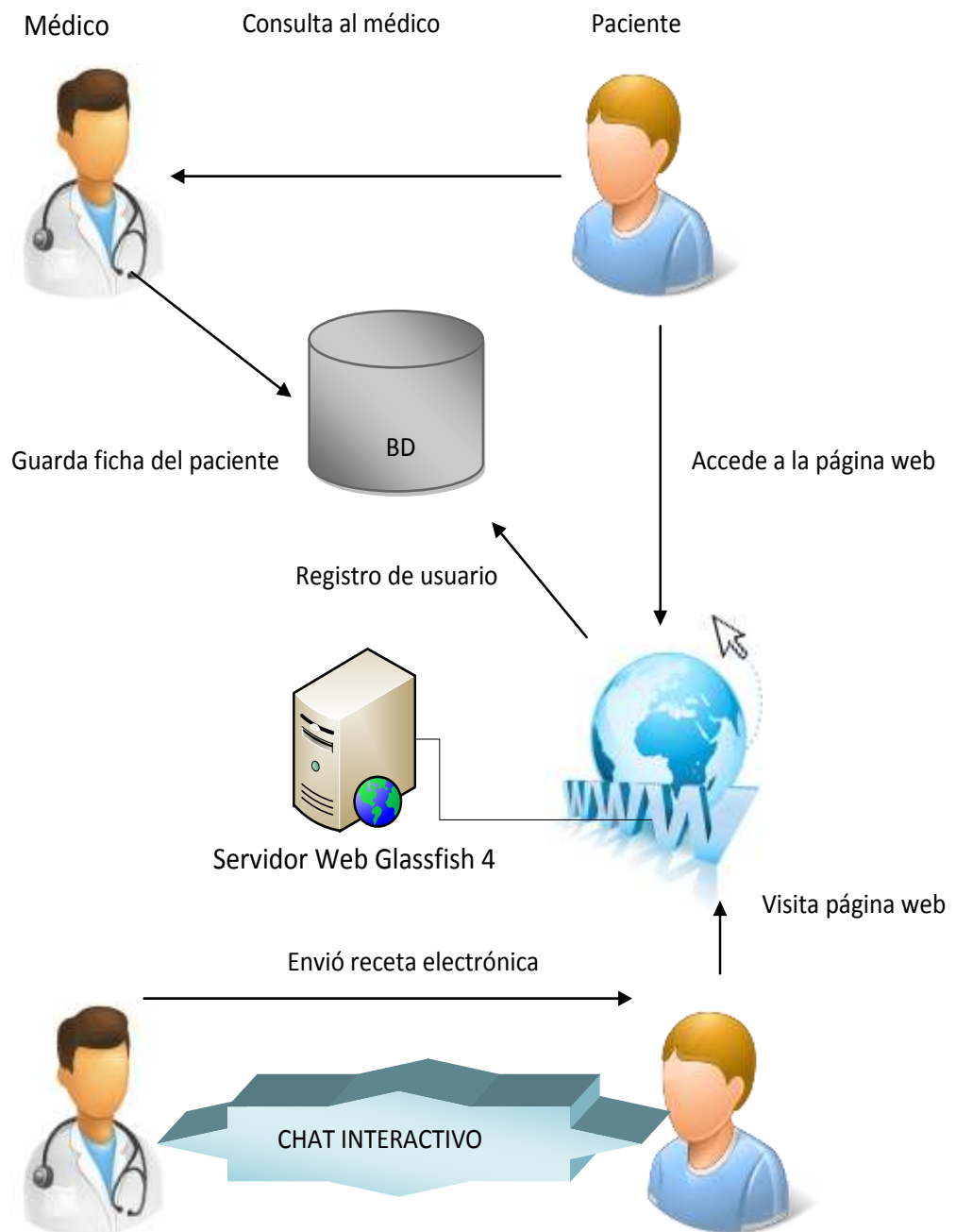


Figura 15 Arquitectura Cliente-Servidor

Elaborado por: Autora

4.1.2. Diseño Arquitectónico

La base de datos creada en la plataforma Mysql, la cual va a permitir tener la información del paciente y del historial médico del mismo.

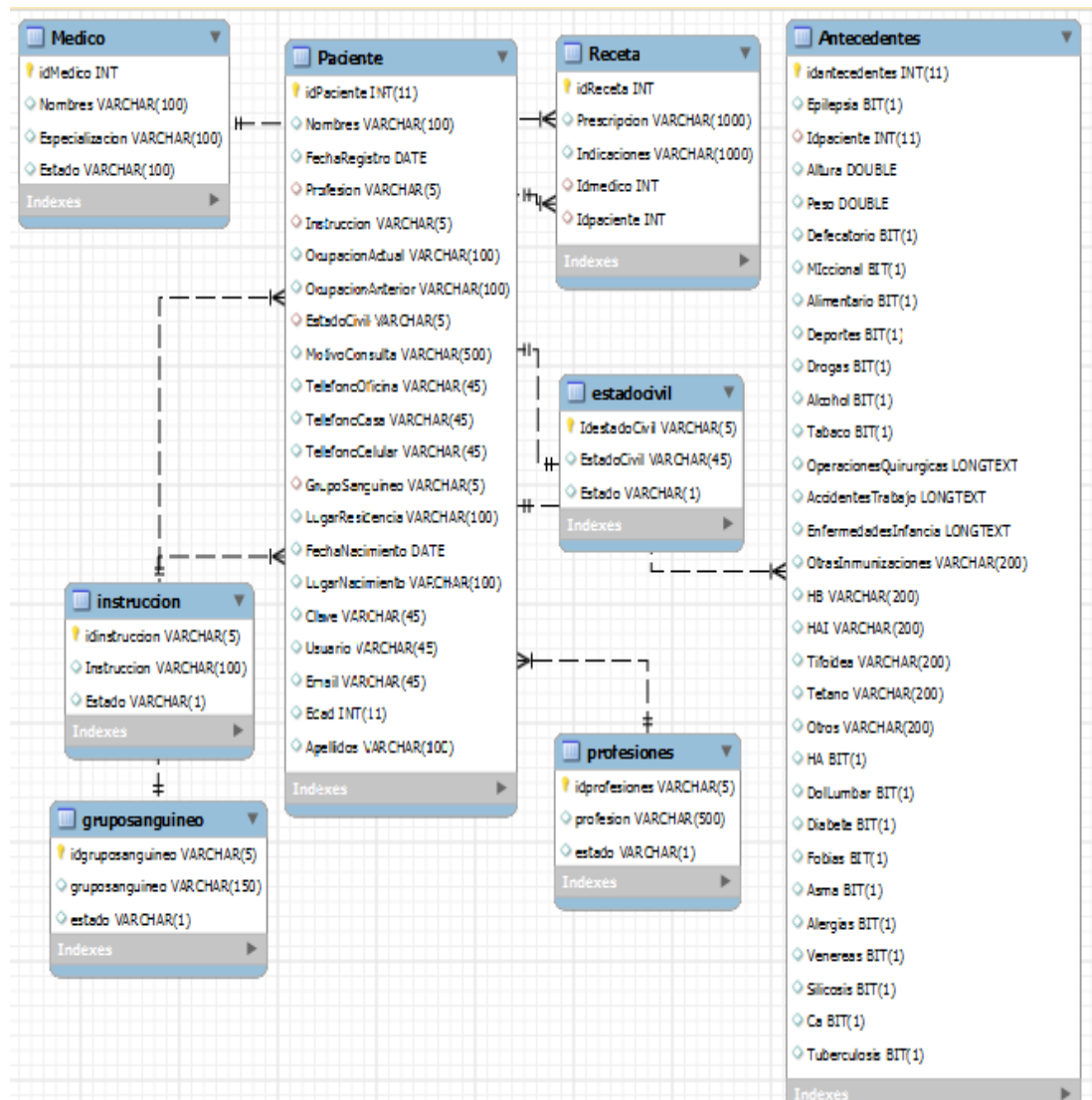


Figura 16 Diseño Arquitectónico Base de Datos

Elaborado por: Autora

4.1.3. Módulos del sistema

El sistema fue dividido en los siguientes módulos para su creación:

- Módulo de Administración Médica
- Módulo de Ingreso de información del paciente

Como el esquema a utilizar es el Modelo-Vista-Controlador (MVC), este serviría de guía para la creación de cada módulo, considerando aquellos objetos que son parte del modelo, de las vistas y controladores de peticiones. La aplicación resultaría más manejable puesto que se sigue el patrón seleccionado para su diseño.

Luego solo habría que identificar aquellas funcionalidades que integran los módulos para finalmente tener el sistema completo.

Se utilizará la arquitectura MVC (Modelo-Vista-Controlador) que es un patrón de arquitectura de software que separa el desarrollo de una aplicación en tres capas:

El siguiente diagrama muestra una representación de dicho modelo:

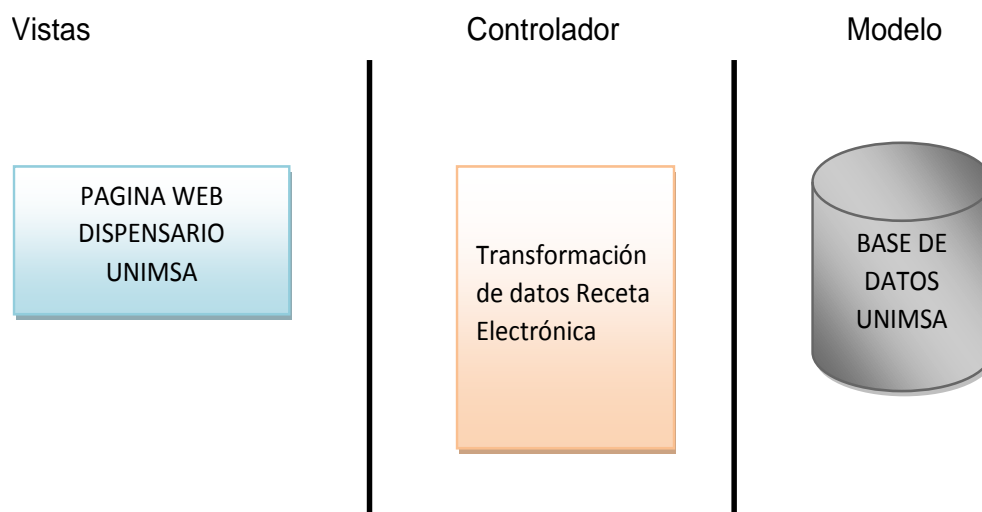


Figura 17 Capas Patrón MVC

Elaborado por: Autora

En donde el modelo encapsula los datos de la aplicación y la lógica del negocio para interactuar con ellos, la vista maneja la interacción con el usuario y la representación del modelo como resultado se tendrá la página del dispensario UNIMSA y el controlador, es el puente entre las dos capas anteriores, y se va a encargar de seleccionar el modelo solicitado por el paciente y seleccionar la vista adecuada para representar.

4.2. Diagrama de clases del Sistema

Diagrama de clase del sistema de receta electrónica:

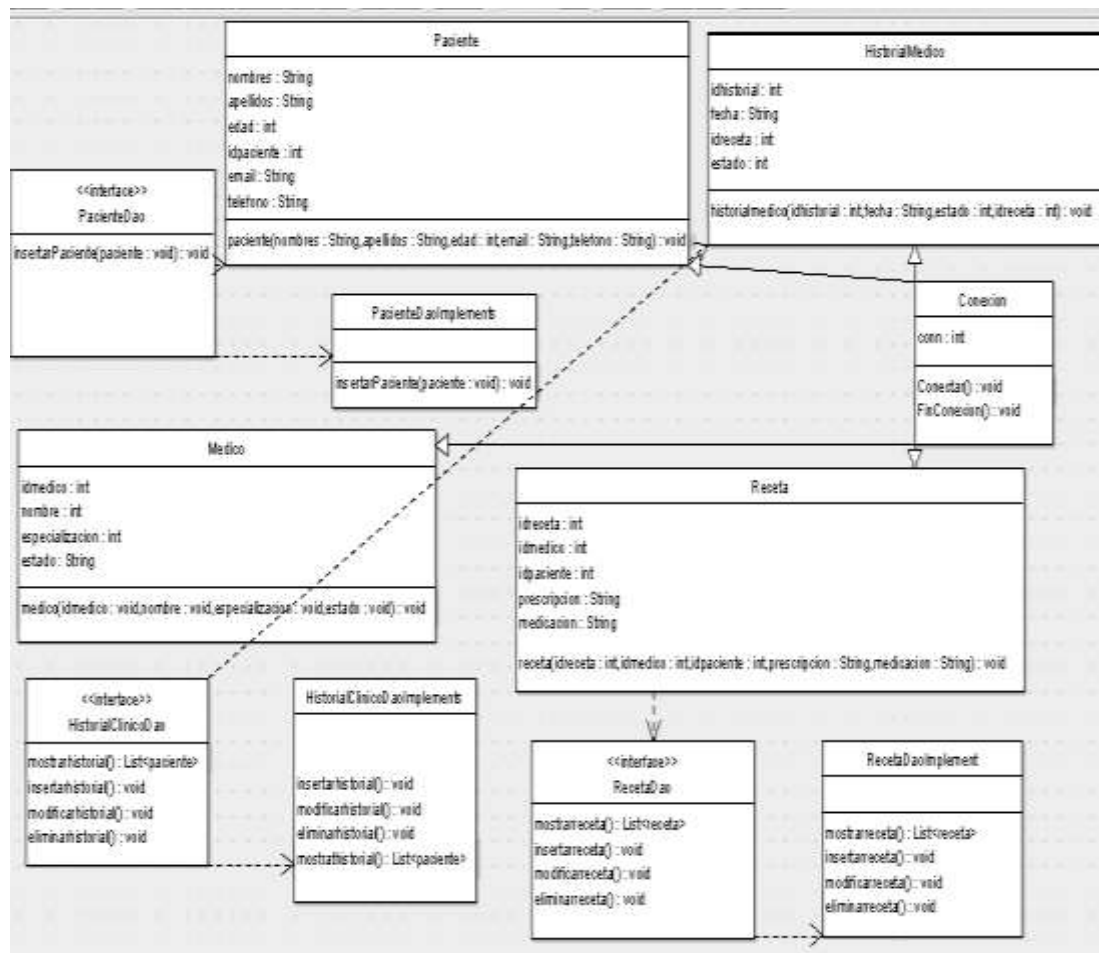


Figura 18 Diagrama de Clases

Elaborado por: Autora

4.2.1. Detalle Diagrama de clases

Tabla 25 Detalles Diagrama de Clases

Clase	Descripción
DATABASE	Esta clase es la que permite conectar a la base de datos a través del método getConnection () y desconectar de la base de datos Close ().
LOGINBEAN	Verifica en la base de datos si existe el paciente, en el caso que exista, direccionará a la página de antecedentes o consultas si es el caso.

UTIL	Permite obtener la sesión y así poder utilizar los objetos de esta clase para invocar métodos de la clase <code>URLConnection</code> cuando sea necesario, como la comprobación de código de estado HTTP, o mensaje de respuesta HTTP.
AUTHFILTER	<code>AuthFilter</code> se llama a través de un <code>RequestDispatcher.include</code> desde el interior del motor de servlets justo antes y justo después de la autenticación y autorización.
ANTECEDENTESBEAN	Clase la cual contiene el método insertar, para poder ingresar los antecedentes en la base de datos.
ESTADOCIVILBEAN	Esta clase es la que contiene el método consultar para obtener el estado civil de la base de datos.
GRUPOSANGUINEOBEAN	Esta clase es la que contiene el método consultar para obtener el grupo sanguíneo de la base de datos.
INSTRUCCIONBEAN	Esta clase es la que contiene el método consultar para obtener las instrucciones de la base de datos.
PACIENTEBEAN	Esta clase es la que contiene el método insertar el cual permite ingresar el paciente en la base de datos.
CLIENTBEAN	Clase que almacena información de mensajes y cuyo método de envío utiliza <code>MailService</code> .

MAILSERVICE	Clase utilizando el API Java Mail y que debe estar configurado con las credenciales de usuario de correo válido.
MESSAGEMANAGER	Clase que emulará una base de datos y se utilizará para la obtención y envío de mensajes.
MESSAGEMANAGERLOCAL	Estas clases representan un singleton start-up Java bean. Esto dará como punto global para acceder a los mensajes. Si desea utilizar la base de datos real, usted debe utilizar un simple EJB sin estado en su lugar.
PROFESIONESBEAN	Esta clase es la que contiene el método consultar para obtener las profesiones de la base de datos.
RECETABEAN	Esta clase es la que contiene el método insertar el cual permite ingresar la receta en la base de datos.
ANTECEDENTES	Contiene los métodos getters y setters para el registro de antecedentes del paciente.
ESTADOCIVIL	Contiene los métodos getters y setters de los estados civiles.
GRUPOSANGUINEO	Contiene los métodos getters y setters de los grupos sanguíneos.
INSTRUCCION	Contiene los métodos getters y setters de las instrucciones.
PACIENTE	Contiene los métodos getters y setters para el registro del paciente.

PROFESION	Contiene los métodos getters y setters de las profesiones.
RECETA	Contiene los métodos getters y setters para el registro de la receta.
MESSAGE	Este es un objeto Java normal que representa un mensaje. Como usted puede notar los implementos clase de interfaz Serializable. Es muy importante para el intercambio de datos entre todas las partes de la aplicación. Cuando se transmite objeto Message debe serializar.
MESSAGEBEAN	<p>Este bean gestionado se usará en la página web para enviar y recibir mensajes. Almacena la fecha del último mensaje que se recibió. Y el cliente solicitará sólo los mensajes nuevos, que aún no fue recibido.</p> <p>Para pasar mensajes al lado del cliente (navegador web) se utilizará el mecanismo de parámetros callback. Los datos son en serie y se transmite en formato JSON.</p>

Elaborado por: Autora

4.3. Modelo lógico de la base de datos

A continuación se especifica el modelo relacional:

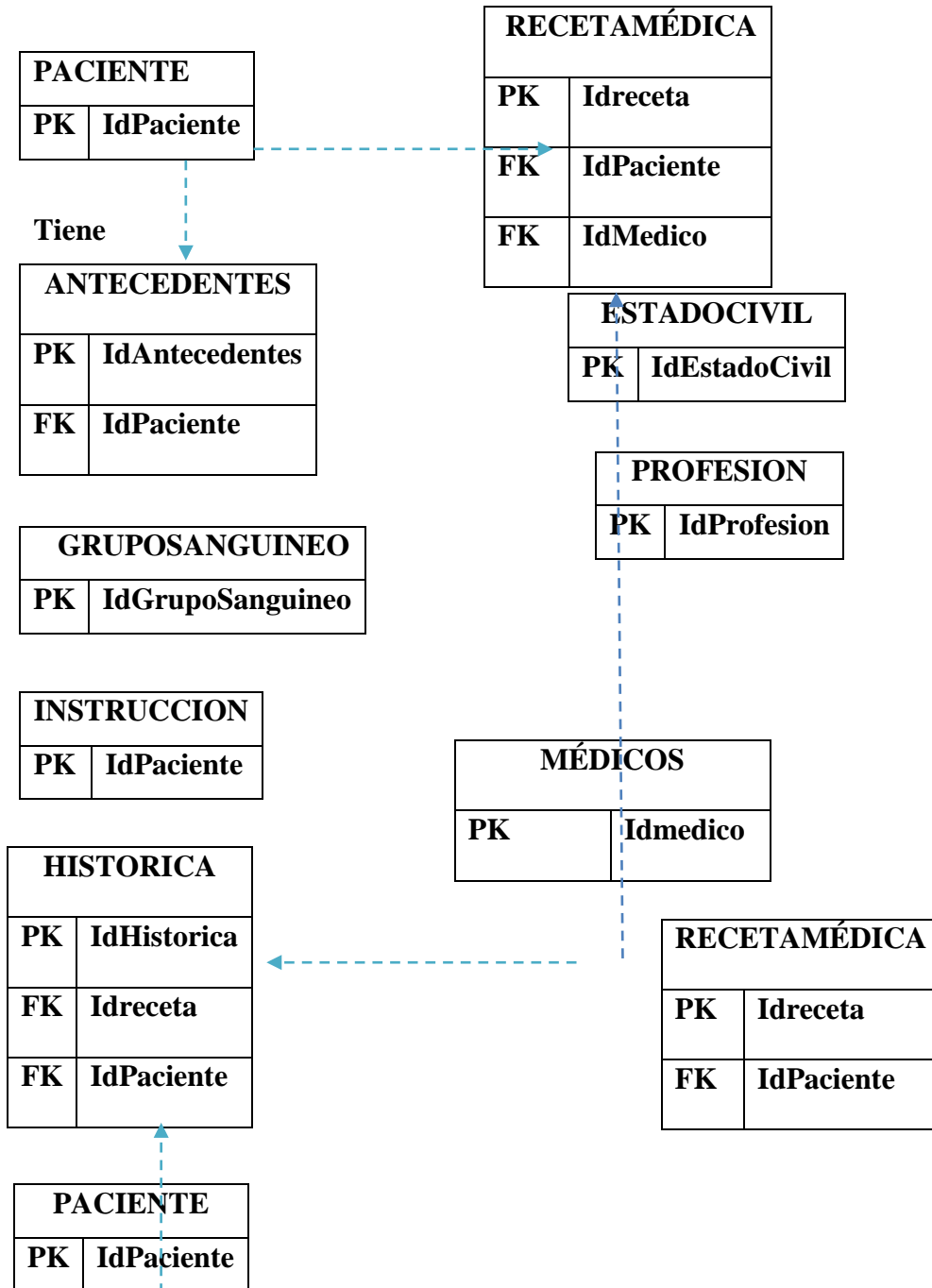


Figura 19 Modelo Lógico BD

Elaborado por: Autora

4.3.1. Código Fuente

Estandarización

Se empezará creando paquetes para tener ordenadas las clases, se usarán los siguientes paquetes:

Persistencia:

En este paquete se guardará las conexiones de hibernate con la base de datos.

Model:

Aquí se guardará los modelos que genere el hibernate con cada una de lass tablas, en clase java y en formato XML.

Bean:

En este paquete se guardará los beans que interactúan con las vistas.

DAO:

Acá se guardará todos los accesos que se realicen a la base de datos.

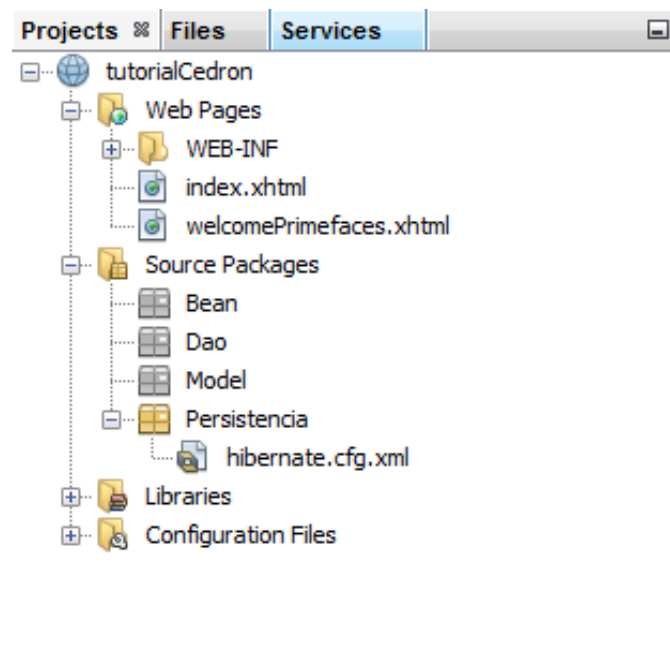


Figura 20 Paquetes

Elaborado por: Autora

Método getConnection():

Se establece una conexión con el origen de datos que representa este objeto MySQLDataSource mediante el nombre de usuario y contraseña especificados.

Si se llama al método getConnection con un nombre de usuario o contraseña que no sean NULL, se reemplazarán las propiedades de contraseña y nombre de usuario que se establecen en la clase.

```
public static Connection getConnection() {
    try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con = DriverManager.getConnection("jdbc:mysql://localhost/unimsa",
            "root", "sorteo");
        return con;
    } catch (Exception ex) {
        System.out.println("Database.getConnection() Error -->" + ex.getMessage());
        return null;
    }
}

public static void close(Connection con) {
    try {
        con.close();
    } catch (Exception ex) {
    }
}
```

Figura 21 Método GETCONECCION

Elaborado por: Autora

Método loginproject():

En este método se obtendrá primero la conexión a la base de datos a través del método getConnection() una vez teniendo la sesión habilitada se podrá enlazar a la página de consulta médica.

```

}

public String loginProject() {
    Connection con = Database.getConnection();
    FacesMessage msg = null;
    boolean result = UserDAO.login(uname, password);
    HttpSession session = Util.getSession();
    session.setAttribute("username", uname);
    int idpaciente;
    System.out.println(result);
    if (result) {
        if ("admin".equals(uname)){
            return "medico";
        }else{
            // get Http Session and store username

            /*HttpSession sessionx = Util.getSession();
            session.setAttribute("username", uname);*/
            try{
                PreparedStatement pre = con.prepareStatement("select count(p.idpaciente) idpaciente from paciente p, antecedentes a

```

Figura 22 Método LOGINPROJECT

Elaborado por: Autora

Método configure():

Se agregará dentro del método configure la ubicación del hibernate.cfg.xml, en este caso "/Persistencia/hibernate.cfg.xml"

```

*/
public class NewHibernateUtil {

    private static final SessionFactory sessionFactory;

    static {
        try {
            // Create the SessionFactory from standard (hibernate.cfg.xml)
            // config file.
            sessionFactory = new AnnotationConfiguration().configure("/Persistencia/hibernate.cfg.xml").buildSessionFactory();
        } catch (Throwable ex) {
            // Log the exception.
            System.err.println("Initial SessionFactory creation failed." + ex);
            throw new ExceptionInInitializerError(ex);
        }
    }
}

```

Figura 23 Método CONFIGURE

Elaborado por: Autora

Métodos DAO:

En clases que terminan en Dao tendrá los siguientes métodos:

- mostrar()
 - Devuelve una lista de objetos receta
- insertar()
- modificar()

- eliminar()

El insertar, modificar y eliminar necesita como parámetro un objeto el cual será modificar, eliminar o insertar.

La función de los métodos es crear una Session de tipo hibernate accediendo al HibernateUtil y se crea un query que devuelve todos los registros de la tabla seleccionada. El lenguaje que usan los queries de Hibernate se llama HQL. Por último se llena una lista de objetos Persona con el resultado del query.

El método de insertar, al igual que el de modificar y eliminar solo variará en la acción de transacción que se realiza: save, update y delete. El formato es muy parecido:

```
] /*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
- */

package Dao;

] import Model.Receta;
- import java.util.List;

] /**
 *
 * @author miguel angel
- */
public interface RecetaDao {
    public void insertarReceta(Receta receta);
    public List<Receta> mostrarRecetas();
    public void modificarReceta(Receta receta);
}
```

Figura 24 Métodos DAO

Elaborado por: Autora

Recetadaoimplement():

Se hará implements a la clase RecetaDao y procederá a modificar los métodos de acceso.

```

public class RecetaDaoImplement implements RecetaDao{
    Connection con = Database.getConnection();
    @Override
    public List<Receta> mostrarRecetas () {
        Session session=null;
        List<Receta> lista=null;
        try{
            session=NewHibernateUtil.getSessionFactory().openSession();
            Query query= session.createQuery("from Receta");
            lista=(List<Receta>) query.list();
        }catch(HibernateException e){
            System.out.println(e.getMessage());
        }
        finally{
            if (session!=null){
                session.close();
            }
        }
        return lista;
    }
}

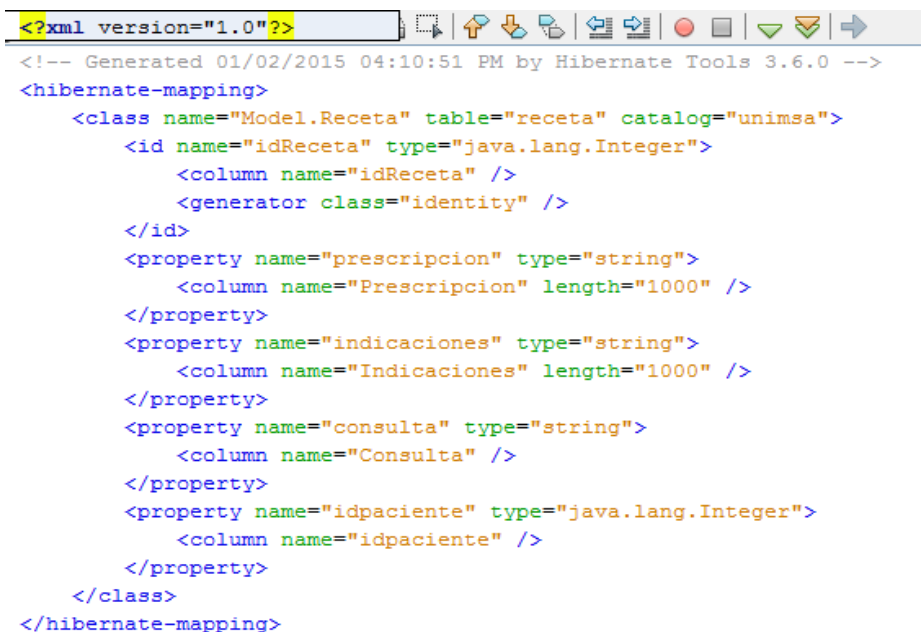
```

Figura 25 RECETADAOIMPLEMENT

Elaborado por: Autora

Clases Model:

Crear las tablas en clase java y en archivo xml que usa hibernate para poder acceder a los registros.



```

<?xml version="1.0"?>
<!-- Generated 01/02/2015 04:10:51 PM by Hibernate Tools 3.6.0 -->
<hibernate-mapping>
    <class name="Model.Receta" table="receta" catalog="unimsa">
        <id name="idReceta" type="java.lang.Integer">
            <column name="idReceta" />
            <generator class="identity" />
        </id>
        <property name="prescripcion" type="string">
            <column name="Prescripcion" length="1000" />
        </property>
        <property name="indicaciones" type="string">
            <column name="Indicaciones" length="1000" />
        </property>
        <property name="consulta" type="string">
            <column name="Consulta" />
        </property>
        <property name="idpaciente" type="java.lang.Integer">
            <column name="idpaciente" />
        </property>
    </class>
</hibernate-mapping>

```

Figura 26 Clases Model

Elaborado por: Autora

Es un mapeo exacto de la base de datos.

```
* Receta generated by hbm2java
*/
public class Receta implements java.io.Serializable {

    private Integer idReceta;
    private String prescripcion;
    private String indicaciones;
    private String consulta;
    private Integer idpaciente;

    public Receta() {
    }

    public Receta(String prescripcion, String indicaciones, String consulta, Integer idpaciente) {
        this.prescripcion = prescripcion;
        this.indicaciones = indicaciones;
        this.consulta = consulta;
        this.idpaciente = idpaciente;
    }
}
```

Figura 27 Clases Model BD

Elaborado por: Autora

hibernaterevenge:

En este xml se ingresa el nombre del esquema de la base de datos y las tablas a la que se van acceder.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-reverse-engineering PUBLIC "-//Hibernate/Hibernate Reverse Engineering DTD 3.0/EN" "http://hibernate.sourceforge.net/hibernate-reverse-engineering.dtd" [ ]>
<hibernate-reverse-engineering>
  <schema-selection match-catalog="unimsa"/>
  <table-filter match-name="receta"/>
</hibernate-reverse-engineering>
```

Figura 28 HIBERNATEREVENGE

Elaborado por: Autora

Metodos bean:

Crear un bean al paquete BEAN, el cual manejará la vista en la cual gestionará las tablas de la base de datos UNIMSA.

En este gráfico muestra que se creará un objeto Receta y luego una lista de objetos receta, a estos objetos se creará getters y setters para que puedan ser accedidos desde la vista. Además se añadirá los métodos insertar, modificar, eliminar y mostrar, los cuales accederán a los métodos establecidos en el DAO.

Dentro del constructor recetaBean() definir lo siguiente: receta = new Receta() para que deje de ser un objeto nulo.

```
@ViewScoped
public class recetaBean implements Serializable {
    private static final long serialVersionUID = 1L;
    private Receta receta;
    private List<Receta> recetas;

    public Receta getReceta() {
        return receta;
    }

    public void setReceta(Receta receta) {
        this.receta = receta;
    }

    public List<Receta> getRecetas() {
        RecetaDao linkdao=new RecetaDaoImplement();
        recetas=linkdao.mostrarRecetas();
        return recetas;
    }

    public void setRecetas(List<Receta> recetas) {
```

Figura 29 Métodos BEAN

Elaborado por: Autora

Metodo send:

El cual retorna un string con el fin de saber si se envió el respectivo mail se le pasa los parametros de la consulta para luego enviar la receta al paciente a través del correo electrónico por medio de una cuenta de google.

```
public String send(int idpaciente, String indicaciones, String prescripcion) {
    statusMessage = "Message Sent";
    Connection con = Database.getConnection();

    String nombres="";
    String email="";
    String mensaje;
    java.util.Date fecha=new Date();
    try{
        //user=paciente.getIdUsuario();
        PreparedStatement pre = con.prepareStatement("select CONCAT(Nombres, ' ', Apellidos) As Nombres,email from paciente");
        pre.setInt(1, idpaciente);
        ResultSet rs = pre.executeQuery();
        while (rs.next()) {
            nombres = rs.getString(1);
            email=rs.getString(2);
        }
    }
}
```

Figura 30 Método SEND

Elaborado por: Autora

Metodo imprimir:

A través de la librería itextpdf se podrá usar lo siguiente:

Se crea el objeto document, esta instancia document representa el documento PDF. Para agregar contenido al documento PDF, se llama a los métodos en el objeto document. El OutputStream es donde el documento PDF generado se escribe. El documento se abre llamando document.open() . Añadir el contenido por medio del Paragraph, y cerrar el documento, para eliminar todo el contenido del documento a la PDFWriter.

Metodo DOFILTER:

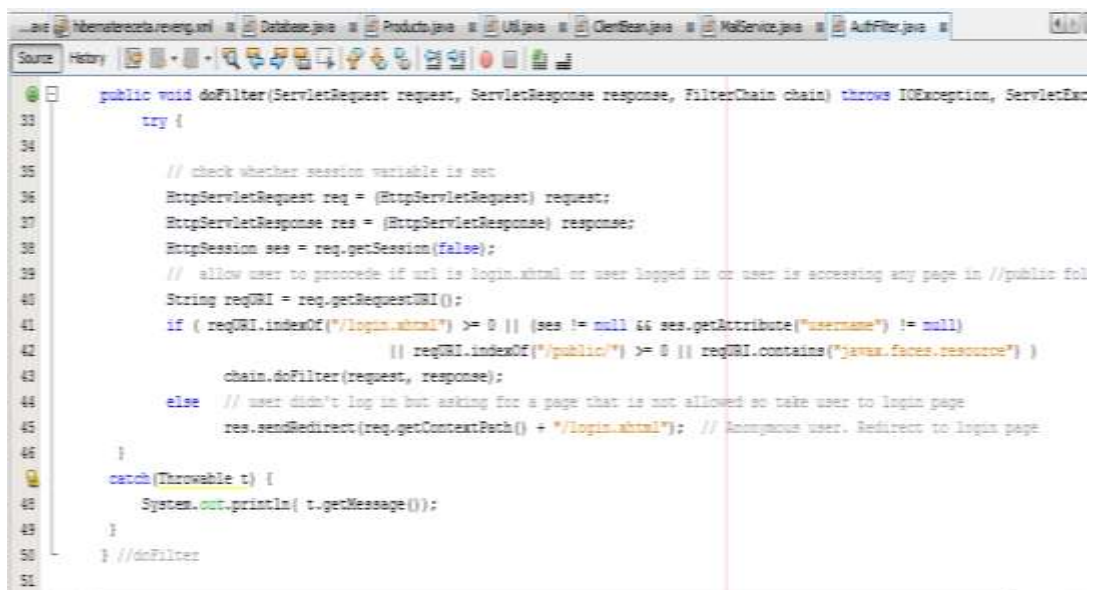
Es un objeto que puede interceptar las peticiones HTTP dirigidas a la aplicación

```
public void imprimir(String indicaciones, String preescrpcion, String idpaciente) {
    try {
        //Generamos el archivo PDF
        String directorioArchivos;
        ServletContext ctx = (ServletContext) FacesContext.getCurrentInstance().getExternalContext().getContext();
        directorioArchivos = ctx.getRealPath("/") + "reports";
        System.out.println(directorioArchivos);
        String name = directorioArchivos + "/document-report.pdf";
        Document document = new Document(PageSize.LETTER.rotate());
        PdfWriter.getInstance(document, new FileOutputStream(name));
        document.open();
        PdfPTable tabledatos = createTableDatos();
        document.add(tabledatos);
        PdfPTable tabledatos2 = createTable2(indicaciones,preescrpcion,idpaciente);
        document.add(tabledatos2);
        /*document.add(new Paragraph("Visita http://rolandopalermo.blogspot.com"));
        document.add(new Paragraph("Nombre: " + getNombre()));
        document.add(new Paragraph("Tipo: " + getTipo()));*/
    }
}
```

Figura 31 Método imprimir

Elaborado por: Autora

web. Con el fin de crear un filtro de servlet debe implementar la javax.servlet.Filter interfaz.



```
public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain) throws IOException, ServletException {
    try {
        // check whether session variable is set
        HttpServletRequest req = (HttpServletRequest) request;
        HttpServletResponse res = (HttpServletResponse) response;
        HttpSession ses = req.getSession(false);
        // allow user to proceed if url is login.xhtml or user logged in or user is accessing any page in //public fol
        String reqURI = req.getRequestURI();
        if ( reqURI.indexOf("/login.xhtml") >= 0 || (ses != null && ses.getAttribute("username") != null)
            || reqURI.indexOf("/") >= 0 || reqURI.contains("/javax.faces.resource") )
            chain.doFilter(request, response);
        else // user didn't log in but asking for a page that is not allowed so take user to login page
            res.sendRedirect(req.getContextPath() + "/login.xhtml"); // Anonymous user. Redirect to login page
    }
    catch(Throwable t) {
        System.out.println( t.getMessage());
    }
} //doFilter
```

Figura 32 Método DOFILTER

Elaborado por: Autora

Un filtro de servlet puede interceptar las peticiones tanto para servlets, JSP, archivos HTML u otro tipo de contenido estático, como se ilustra en el siguiente diagrama:

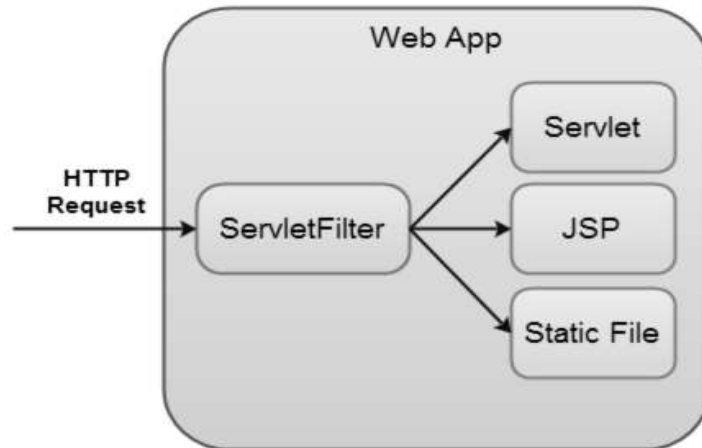


Figura 33 Filtro SERVLET

Elaborado por: Autora

Métodos envío de Mensajes:

Para el envío de correo se utiliza la librería Java Mail API. Sending e-mail, con sólo un protocolo (SMTP) para manejar.

```
...ava AuthFilter.java UserDAO.java MessageBean.java MessageManager.java MessageManagerLocal
Source History
26 @Singleton
27 @Startup
28 public class MessageManager implements MessageManagerLocal {
29
30     private final List messages =
31         Collections.synchronizedList(new LinkedList());
32
33     @Override
34     public void sendMessage(Message msg) {
35         messages.add(msg);
36         msg.setDateSent(new Date());
37     }
38
39     @Override
40     public Message getFirstAfter(Date after) {
41         if(messages.isEmpty())
42             return null;
43         if(after == null)
44             return (Message) messages.get(0);
45         for (Iterator it = messages.iterator(); it.hasNext();) {
46             Message m = (Message) it.next();
```

Figura 34 Send Message

Elaborado por: Autora

```

* @param recipient Internet address of the recipient
* @param subject the subject of the message
* @param message the message
* @param idpaciente
* @throws MessagingException
*/
public static void sendMessage(String recipient, String subject, String message, int idpaciente) throws MessagingException {

    if ( theService == null ) {
        theService = new MailService();
    }
    String directorioArchivos;
    ServletContext ctx = (ServletContext) FacesContext.getCurrentInstance().getExternalContext().getContext();
    directorioArchivos = ctx.getRealPath("/") + "img";
    String ruta=directorioArchivos+"\\logo.jpg";
    MimeMessage mimeMessage = new MimeMessage(mailSession);

    // This HTML mail have to 2 part, the BODY and the embedded image
    //

```

Figura 35 Método de envío de mensaje

Elaborado por: Autora

Estos métodos sirven para almacenar la fecha de los mensajes que reciben para el chat interno. Y el cliente solo podrá ver los nuevos mensajes, ya que plantea como una especie de cliente servidor.

```

}

public void firstUnreadMessage(ActionEvent evt) {
    RequestContext ctx = RequestContext.getCurrentInstance();

    Message m = mm.getFirstAfter(lastUpdate);

    ctx.addCallbackParam("ok", m!=null);
    if(m==null)
        return;

    lastUpdate = m.getDateSent();

    HttpSession sessionx = Util.getSession();
    String user=sessionx.getAttribute("username").toString();

    ctx.addCallbackParam("user", m.getUser().toString());
    ctx.addCallbackParam("dateSent", m.getDateSent().toString());
    ctx.addCallbackParam("text", m.getMessage());
}

```

Figura 36 Método de envío de mensaje 2

Elaborado por: Autora

CAPITULO 5

IMPLEMENTACIÓN Y PRUEBAS

Para la implementación del sistema de recetario electrónico se alquila un servidor en donde se instala el servidor web y la base de datos para poder tener una salida al internet, por medio de la vía ftp conectar al servidor del proveedor y se pasa los fuentes el WAR, extensiones, la base de datos etc. Los componentes que se utilizan son los siguientes:

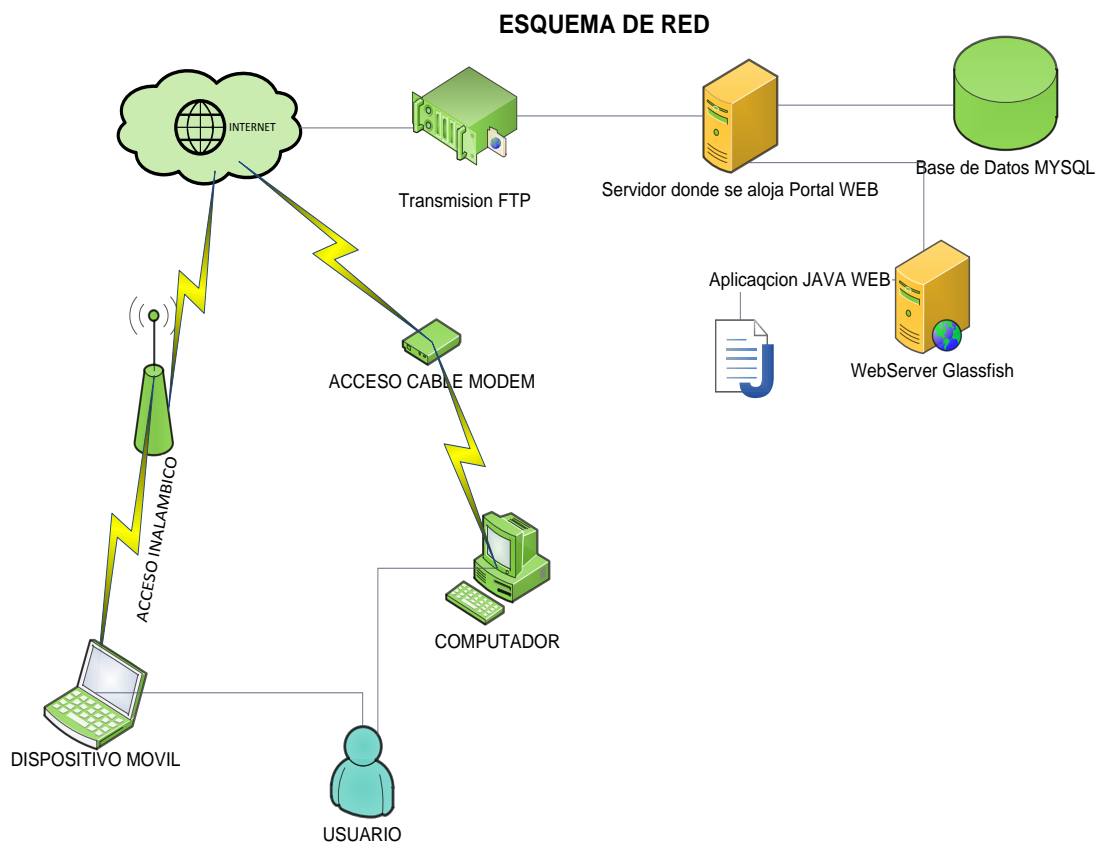


Figura 37 Esquema de Red

Elaborado por: Autora

En la figura 37 se muestra el esquema de red en donde el paciente o médico se conecta al internet para acceder al portal web el cual contiene una base de datos mysql en donde encuentra los datos en brutos y este portal a su vez se encuentra sujeto al servidor de aplicaciones.

El equipo donde se aloja el portal web es un servidor con las siguientes características:

- 512MB de memoria
- 1 Core de procesador
- 20GB de disco duro
- 1TB de transferencia

Tiene:

- Base de datos: Mysql
- Java: JDK 1.7
- Servidor de aplicaciones: Glassfish 4

A este equipo se instaló el JDK 1.7 que es el que permite compilar la aplicación java, el servidor de aplicaciones de código abierto Glassfish versión 4 y la base de datos mysql. Se restauró la base de datos llamada UNIMSA y por último a través de un servidor ftp pasan los compilados al servidor para levantar la aplicación web.

5.1. Capas del Sistema y Comunicación entre Capas

Las capas del sistema de recetario electrónico que interactúan entre sí son las siguientes:

- Capa de datos
- Capa de negocio
- Capa de presentación

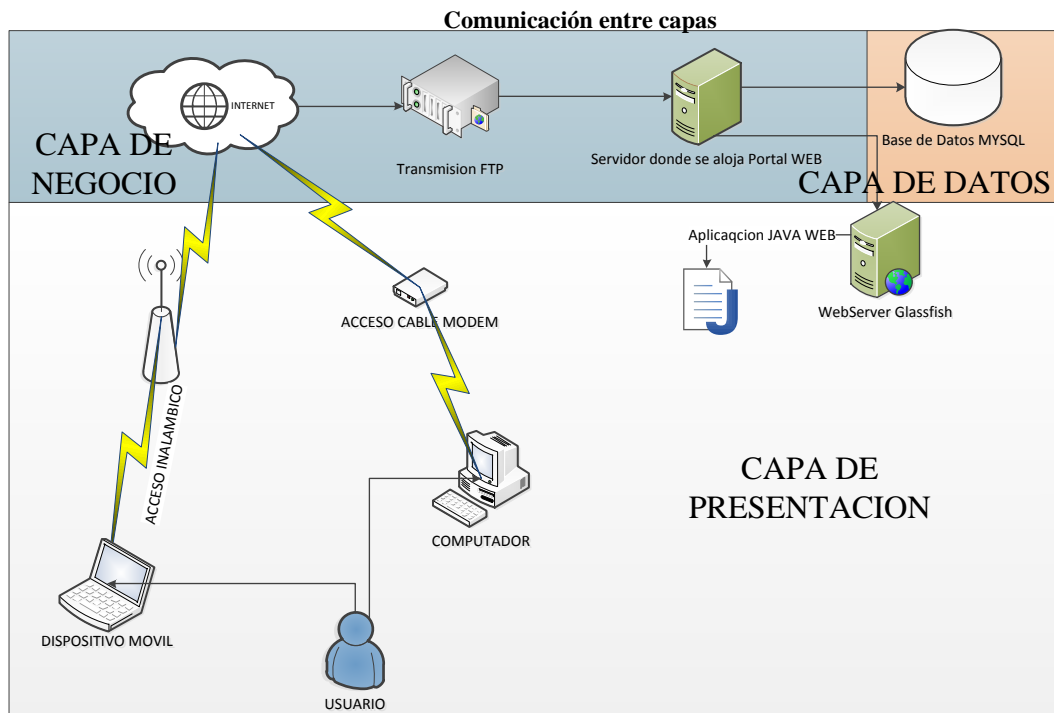


Figura 38 Comunicación entre capas del sistema

Elaborado por: Autora

5.1.1. Capa de datos

Esta capa es la que permite acceder a los datos ya sea desde la base de datos del portal web o de cualquier servicio web.

5.1.2. Capa de negocio

Es la capa en donde realiza el procesamiento de la información, se recibe cualquier petición y se envía la respuesta deseada.

5.1.3. Capa de presentación

Es la interfaz que interactúa con el usuario presentando a este todas las funcionalidades del sistema de manera fácil.

5.2. Plan de Pruebas

5.2.1. Pruebas unitarias

Tabla 26 Prueba Unitaria Ingreso al Portal

Caso de Uso: Ingreso al Portal Web		N°: 1	
Escenario: Ingreso del usuario y la clave (acceso denegado)			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones	No estar registrado en el sistema		
Datos de Entrada	usuario, clave		
Descripción de Pasos	Ingrese el usuario y la clave Presionar el botón ingresar		
Resultado Esperado	Mensaje “El usuario no se encuentra registrado”	Cumplimiento	<input type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

Tabla 27 Prueba Unitaria Ingreso al Portal (Éxito)

Caso de Uso: Ingreso al Portal Web		N°: 2	
Escenario: Ingreso del usuario y la clave (acceso exitoso)			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones	Registrar paciente		
Datos de Entrada	usuario, clave		
Descripción de Pasos	Ingrese el usuario y la clave Presionar el botón ingresar		
Resultado Esperado	Redirección a la página de Antecedentes	Cumplimiento	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

Tabla 28 Prueba Unitaria Registro de Paciente

Caso de Uso: Registro de Paciente		N°: 3	
Escenario: Registro de Paciente			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones	Ingresar los datos que son obligatorios		
Datos de Entrada	usuario, clave		
Descripción de Pasos	Ingrese todo los datos obligatorios Presionar el botón guardar		
Resultado Esperado	Datos guardados con éxito en la base de datos en la tabla pacientes	Cumplimiento	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

Tabla 29 Prueba Unitaria Registro de Paciente

Caso de Uso: Registro Antecedentes		N°: 4	
Escenario: Ingresar los antecedentes del paciente			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones			
Datos de Entrada			
Descripción de Pasos	Ingresar los datos necesarios Presionar el botón guardar		
Resultado Esperado	Se guarda los antecedentes en la base de datos en la tabla antecedentes	Cumplimiento	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

Tabla 30 Prueba Unitaria Consulta Médica

Caso de Uso: Consulta Medica		Nº: 5	
Escenario: Ingreso de los síntomas que le aquejan al paciente			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones	Registrar los antecedentes del paciente		
Datos de Entrada	Síntomas		
Descripción de Pasos	Ingreso de síntomas que aquejan al paciente Presionar en el botón Guardar		
Resultado Esperado	Guardará en la base de datos los síntomas del paciente en tabla de consulta médica	Cumplimiento	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

Tabla 31 Prueba Prescripción Médica

Caso de Uso: Prescripción medica		N°: 6	
Escenario: Prescripción medica			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones	Ingrese los síntomas		
Datos de Entrada	Prescripción, recomendaciones		
Descripción de Pasos	Presionar en el botón recetar Ingresar los datos Presionar el botón guardar		
Resultado Esperado	Guarda en la base de datos la receta médica del paciente	Cumplimiento	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

Tabla 32 Prueba envió de receta médica

Caso de Uso: Prescripción medica		N°: 7	
Escenario: Envío de Receta Medica			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones	Ingrese los síntomas		
Datos de Entrada			
Descripción de Pasos	Presionar en el botón enviar receta		
Resultado Esperado	La receta llegue al correo electrónico del paciente	Cumplimiento	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

Tabla 33 Prueba imprimir receta

Caso de Uso: Prescripción medica		N°: 8	
Escenario: Imprimir receta			
Responsable: Pamela Betancourt Yépez		Fecha: 26/02/2015	
Precondiciones	Prescripción médica del médico		
Datos de Entrada			
Descripción de Pasos	Presionar en el botón imprimir receta		
Resultado Esperado	Visualizar la receta para imprimirla	Cumplimiento	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO
Resultado Obtenido	Errores:	Fallas Provocadas:	
Recomendación u Observación			

Elaborado por: Autora

5.3. Resultados de las Pruebas y métricas tomada

De cada una de las pruebas realizadas se procede a dividirla en tres partes que son la funcionalidad, complejidad y el tiempo de respuesta, a cada una de estos factores se le asigna un valor para poder medir el resultado de cada escenario de prueba. Entre mayor sea el valor de funcionamiento quiere decir que la prueba fue exitosa.

Si el valor de complejidad es mayor quiere decir que la prueba para el usuario fue compleja y entre mayor tiempo de respuesta se muestre quiere decir que al momento de realizar la prueba fue muy rápida la obtención de la respuesta.

Tabla 34 Resultado de Pruebas

Escenario de pruebas	Funcionamiento	Complejidad	Tiempo de respuesta
Ingreso al Portal Web	100	20	60
Registro del Paciente	100	20	70
Registro Antecedentes	100	10	70
Consulta Médica	100	20	80
Prescripción Médica	100	10	80

Elaborado por: Autora

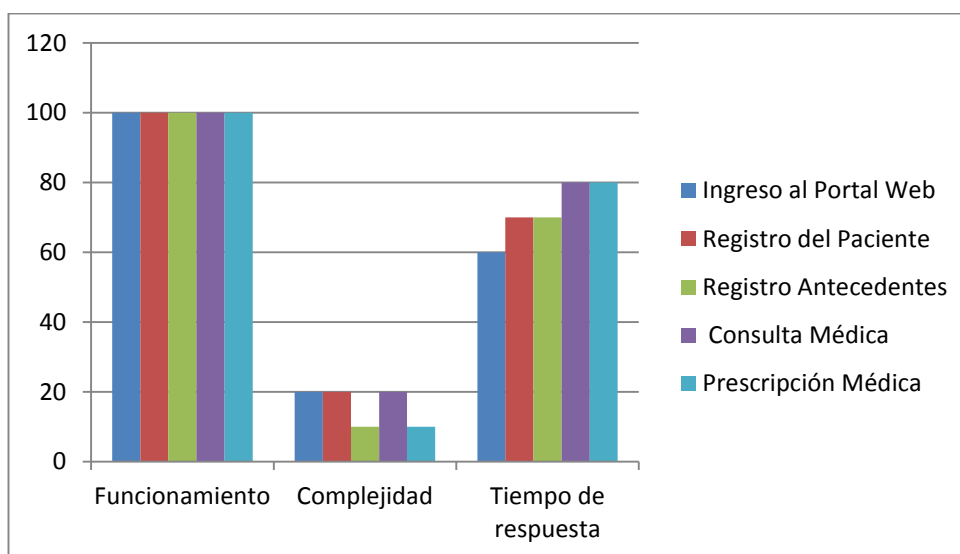


Figura 39 Resultado de pruebas

Elaborado por: Autora

Concluye las pruebas realizadas fueron exitosas y su tiempo de respuesta fue muy rápido a demás que es de muy fácil manejo el portal web.

CAPITULO 6

CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

- Como resultado del desarrollo de este proyecto de tesis se tendrá un sistema de información que permite generar una prescripción médica para mejorar el estado de salud de una persona, cumpliendo con ciertas restricciones de tiempo, requerimientos y especificaciones del médico del dispensario UNIMSA.
- Un aspecto importante para que el programa sea funcional es la manera de interactuar el médico y el paciente ya que sin esta comunicación no tendrá una solución para cualquier enfermedad ambulatoria.
- El tiempo de espera de la prescripción médica depende del médico encargado, de ver el estado del paciente mediante la descripción de los síntomas y los antecedentes que posee el paciente para proceder a enviar la receta a la persona que necesite de este proceso médico.
- El desarrollo del sistema de esta tesis requiere del almacenamiento de la información en una base de datos MySQL y la implementación de un Modelo de datos vista controlador, lo que permite interactuar de una manera fácil y rápida en los procesos de Ingreso, Eliminación, Actualización y Consultas de los datos.
- La creación de la interfaz del sistema se desarrollará bajo el entorno del lenguaje de programación JAVA; pero pudo desarrollarse en cualquier lenguaje de alto nivel.

6.2. Recomendaciones

- Se desea que para un futuro, se incluyan nuevos requerimientos y actualizaciones del presente proyecto; en apoyo del médico tratante.
- Incluir la manera de asociar la receta electrónica a una distribuidora farmacéutica y que el paciente pueda recibir inmediatamente los medicamentos que fueron prescritos por el médico.
- Implementar una capa adicional a las vistas y al modelo para tener el sistema en varios idiomas, actualmente solo soporta el español porque se creará para Latinoamérica.
- Establecer métricas para medir la calidad del producto de software que se está desarrollando.
- Por último, adaptar el sistema a una red de dispensarios médicos y cobrar un porcentaje por la licencia del mismo, ya que esto es un avance para la medicina.

6.3. Bibliografía

Holzner, S. (2000). *la biblia de java 2*. MADRID: anaya multimedia.
Groussard, T. (2004). *java enterprise edition*. MADRID: eni.
Hlavats, I. (2013). *starter primefaces*. EEUU: packt publishing.
Prasat, S.(2013).*primefaces beginner's guide*. EEUU: packt publishing.

Vera Santana, & Lara Vásquez, R. (2014). *dspace.espol.edu.ec*. Recuperado el 2015, de
<https://www.dspace.espol.edu.ec/bitstream/123456789/9105/1/Implementacion%20de%20un%20portal%20web%20para%20la%20automatizaci%C3%B3n%20del%20proceso%20de%20consultor%C3%ADas.pdf>

Mesa, J., & Torres, R. (2012). *univalle*. Recuperado el 2015, de
http://eisc.univalle.edu.co/materias/Material_Desarrollo_Software/expoDAO.pdf

Referencias web:

ChauhaanNagesh. (2014). *primefaces-framework*. Recuperado el 2015, de
primefaces-framework: <http://www.genbetadev.com/frameworks/primefaces-framework-sobre-jsf-2-0-primeros-pasos>

dis.um.es. (2 de 2 de 2013). *dis.um.es*. Obtenido de dis.um.es:
http://dis.um.es/~lopezquesada/documentos/IES_1213/IAW/curso/UT5/ActividadesAlumnos/10/index.htm

GabrielGonzalez. (2013). *javaserver-faces*. Recuperado el 2015, de javaserver-faces:
<https://kalistog.wordpress.com/javaserver-faces-jsf/>

GabrielGonzalez. (2013). *kalistog.wordpress*. Recuperado el 2015, de
<https://kalistog.wordpress.com/ajax-y-javafx/>

HOLZNER, S. (2000). *LA BIBLIA DE JAVA 2*. MADRID: ANAYA MULTIMEDIA.

jsf. (16 de 05 de 2011). *jsf*. Obtenido de jsf:
<https://kalistog.wordpress.com/javaserver-faces-jsf/>

jvm. (15 de 03 de 2010). *jvm*. Obtenido de jvm: <http://gl-eqn-programacion-ii.blogspot.com/2010/03/jvm-jdk-jre-conceptos-fundamentales-de.html>

jvm-jdk. (15 de 03 de 2010). *jvm-jdk*. Obtenido de jvm-jdk: <http://gl-e-pn-programacion-ii.blogspot.com/2010/03/jvm-jdk-jre-conceptos-fundamentales-de.html>

kalistog. (5 de 06 de 2013). *kalistog*. Obtenido de kalistog: <https://kalistog.wordpress.com/ajax-y-javafx/>

Nagesh, C. (2013). *oodlestechnologies*. Recuperado el 2015, de <http://www.oodlestechnologies.com/blogs/Introduction-and-Architecture-of-Hibernate-4>

Oodles. (2009). *oodlestechnologie*. Recuperado el 2014, de oodlestechnologie: <http://www.oodlestechnologies.com/blogs/Introduction-and-Architecture-of-Hibernate-4>

oodlestechnologies. (2 de 01 de 2014). *oodlestechnologies*. Obtenido de oodlestechnologies: <http://www.oodlestechnologies.com/blogs/Introduction-and-Architecture-of-Hibernate-4>

oodlestechnologies. (6 de 05 de 2014). *oodlestechnologies*. Obtenido de oodlestechnologies: <http://www.oodlestechnologies.com/blogs/Introduction-and-Architecture-of-Hibernate-4>

prezi. (s.f.). *prezi*. Obtenido de prezi: <https://prezi.com/hgqqnxi6end/copy-of-unidad-2-arquitectura-de-aplicaciones-en-dispositivos-movil/servicios-web>

servicios-web. (2 de 06 de 2013). *servicios-web*. Obtenido de servicios-web: <https://kalistog.wordpress.com/servicios-web/>

tecnologia. (2 de 2 de 2014). *tecnologia*. Obtenido de tecnologia: <http://www.significados.com/tecnologia/>

training.genexus. (s.f.). *training.genexus*. Obtenido de training.genexus: <http://training.genexus.com/files/despricion-basica-de-conceptos-sobre-aplicaciones-web?es>

Anexos

Anexo A Manual de usuario del Recetario Electrónico

El propósito de este manual es facilitar al usuario la operación de las diferentes páginas web y opciones que contiene este sistema, así como las funciones de los botones de los mismos.

De manera tal que, el usuario podrá obtener información necesaria para las distintas herramientas que ofrece este sistema web.

Entre las características que ofrece este sistema Web, se especifican las siguientes:

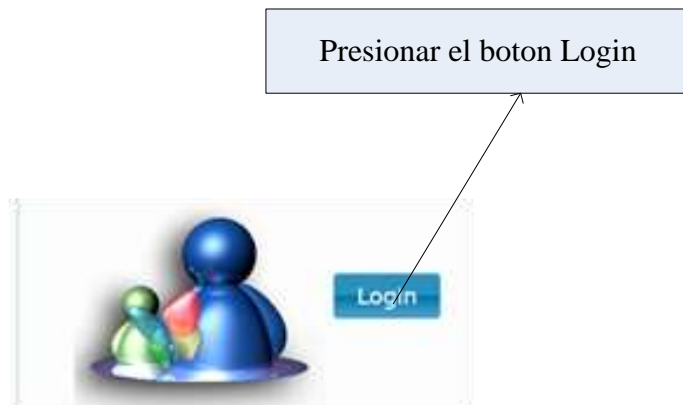
- Es amigable y de fácil manejo, ya que queda a conveniencia del usuario utilizar el Mouse o el Teclado.
- Las personas obtienen una prescripción médica desde la comodidad de su hogar.
- Las personas pueden interactuar entre si y pueden hacerles preguntas al médico.

Ingreso al portal web

Una vez que se accede al sitio se mostrará la pantalla de Login. Como se muestra en la siguiente imagen.



En esta página se ingresa el usuario y la clave, para luego dar click en el botón “Login” para acceder al portal web.



Registro de paciente en el portal web

En el caso de que aparezca el mensaje de que el paciente no está registrado, es porque el paciente necesita previamente registrar sus datos, para hacer esto se tiene que presionar en el botón “Registrarse”.



Para que un usuario tenga acceso al sistema primero debe de ser registrado, para ello es el siguiente formulario de registro de paciente.

Registro del Paciente

Nombres*: Apellidos*:

Edad*: Lugar Nacimiento*:

Fecha Nacimiento*: Lugar Residencia*:

Grupo Sanguineo: Estado Civil:

Instruccion: Profesion:

Teléfono Casa: Teléfono Celular:

Teléfono Trabajo: Motivo Consulta:

Email*: Ocupación Actual:

Ocupación Anterior: Usuario*:

Clave*:

En donde debemos ingresar los siguientes datos:

Nombre del campo	Descripción
Nombres	Nombres del paciente
Apellidos	Apellidos del paciente
Edad:	Edad actual del paciente
Lugar de nacimiento	Lugar de nacimiento del paciente
Fecha de Nacimiento	Fecha de nacimiento del paciente
Lugar Residencia	Lugar en donde vive el paciente
Grupo Sanguíneo	Tipo de sangre del paciente
Estado Civil	Estado civil del paciente
Profesión	Profesión que desempeña el paciente
Instrucción	Instrucción del paciente
Teléfono de casa	Teléfono de la casa del paciente
Teléfono celular	Teléfono celular del paciente
Teléfono del trabajo	Teléfono del trabajo del paciente
Motivo consulta	Motivo de la consulta que va a realizar al médico
Email	Dirección de correo electrónico
Ocupación actual	Ocupación actual del paciente
Ocupación anterior	Ocupación anterior del paciente

Usuario	Usuario con el cual va a ingresar al sistema
Clave	Clave para el ingreso del sistema

Cuando se presiona en el botón “Enviar” los datos serán guardados en la base de datos y pse podrá ingresar en el sistema de recetario electrónico.



Registro de antecedentes

Una vez registrado el paciente podrá ingresar a la siguiente página, en la cual el paciente deberá llenar el formulario de antecedentes clínicos a continuación se muestra la página web de antecedentes.

Formulario de Ingreso de Antecedentes

Antecedentes Personales:

Epilepsia: Tuberculosis: Ca: Síncopa:

Venereas: Asma: Alergias: Fobias:

Diabetes: Dol Lumbar: H.A.: Otros:

Inmunizaciones

Tetano: Tifoidea:

H.A.: H.B.:

Otras:

Enfermedades de la Infancia:

Accidentes de Trabajo:

En donde el paciente debe seleccionar de qué enfermedades ha sufrido, además debe ingresar si es que ha sido inmunizado de: tétano, tifoidea, hepatitis A y hepatitis B.

Antecedentes Personales:

Epilepsia: Tuberculosis: Ca: Síncopa:

Venereas: Asma: Alergias: Fobias:

Diabetes: Dol Lumbar: H.A.: Otros:

Inmunizaciones

Tetano: Tifoidea:

H.A.: H.B.:

Otras:

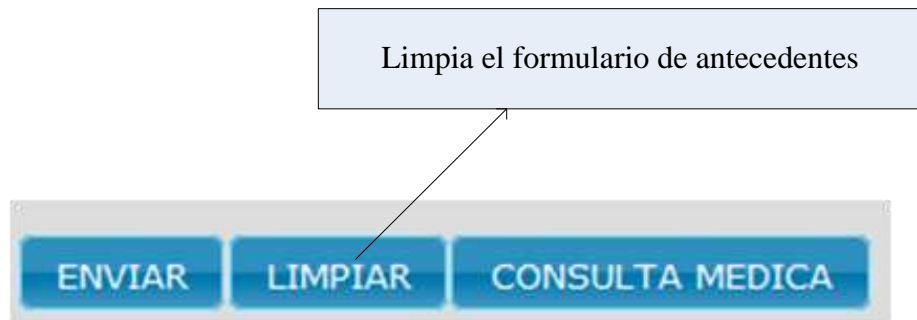
El paciente debe ingresar todas las enfermedades que ha tenido en la infancia, los accidentes laborales y las operaciones quirúrgicas que ha tenido.

En esta página web el paciente también debe seleccionar los hábitos que tiene en su vida.

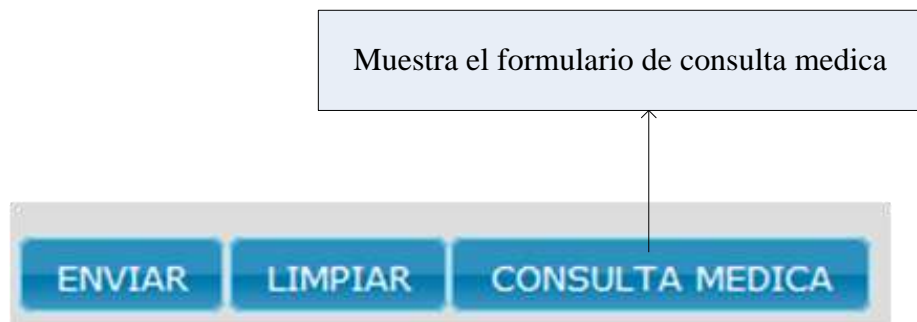
Y por último el paciente ingresará su peso y altura.

Una vez llenado el formulario el paciente deberá dar clic en el botón “Enviar” y todos estos datos se guardan automáticamente en la base de datos.

Si se presiona el botón “Limpiar”, este tendrá la función de limpiar el formulario de antecedentes.



Si se presiona el botón de consulta médica el paciente podrá enlazarse a la página web de consulta al médico en donde se podrá ingresar los síntomas que le aqueja al paciente y además podrá chatear con el médico, este botón solo funciona cuando el paciente tiene ingresado sus antecedentes.



Consulta medica

En el caso de que el paciente ya tenga registrado y tenga ingresado la ficha de antecedentes, este podrá realizar una consulta médica en donde podrá ingresar lo que le aqueja para luego recibir un tratamiento médico.

Consulta

Describe lo que siente:

me duele la barriga

Guardar Consulta

LIMPIAR

Escribe tu Alias:

Mensaje:

LIMPIAR Enviar!

Chat

En este formulario de consulta médica se ingresa todos los síntomas que tiene el paciente, es decir el paciente va a detallar todo los síntomas desde cuando el paciente comienza a sentirse mal y por último se presiona el botón “Guardar consulta” para que esto quede registrado en la base de datos.

Consulta

Describe lo que siente:

me duele la barriga

Presionar el botón guardar consulta

Guardar Consulta

LIMPIAR

Si presionamos el botón limpiar se borrarán los datos que hemos ingresado en el formulario de consulta médica.

En el chat del sistema de receta electrónica sirve para contactarse con el médico con el fin que el paciente pueda hacer cualquier tipo de consulta que necesite y por ende el doctor le contestará y emitirá la receta.

Escribe tu Alias:

Mensaje:

Chat

[Fri Mar 27 00:01:45 COT 2015] mmm: hola

UNIMSA-2015

Para proceder con el chat interno el paciente deberá ingresar un alias que es un nombre o un apodo que se atribuye al paciente, se debe escribir el mensaje que desea que llegue al médico y se da clic en el botón enviar.

Escribe tu Alias:

Mensaje:

Chat

[Fri Mar 27 00:01:45 COT 2015] mmm: hola

- Nombre que aparecerá en el mensaje
- Mensaje que se envía al medico
- Se presiona este botón para enviar el mensaje
- Lugar donde va aparecer los mensajes enviados y recibidos

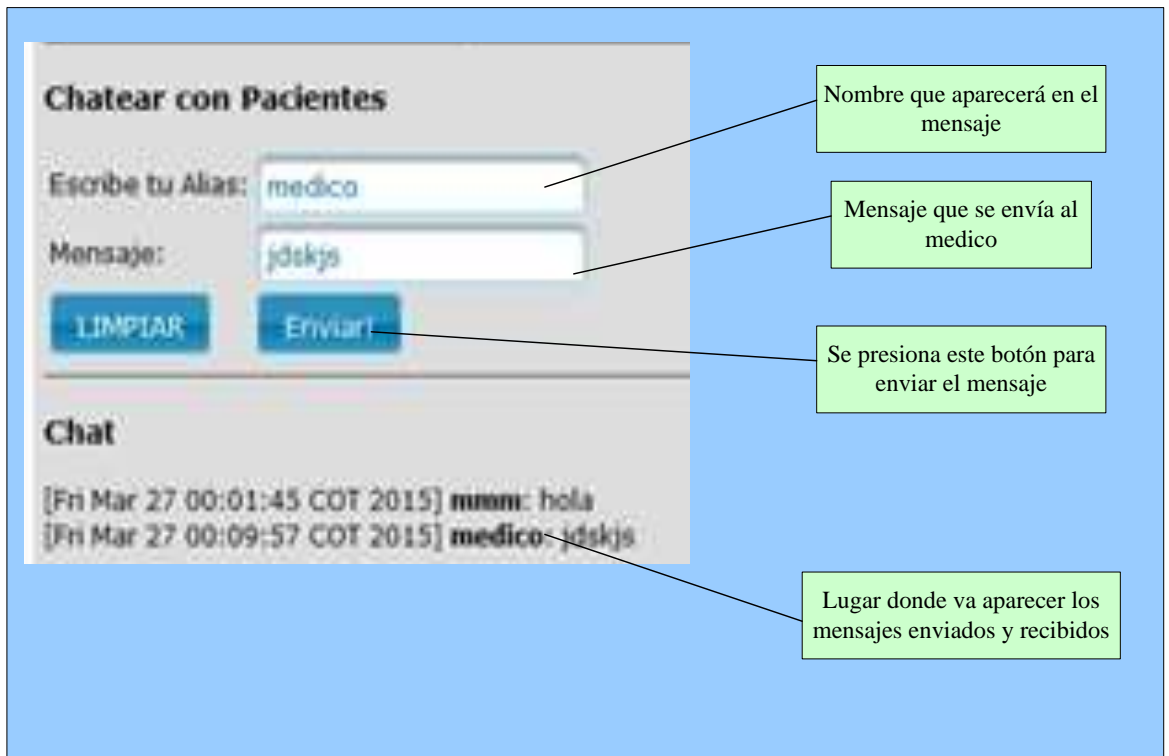
Prescripción y envió de receta medica

En el caso de que ingrese el doctor al sistema de recetario electrónico con su respectivo usuario y clave se abrirá la pantalla en la cual podrá verificar a los pacientes que han enviado consultas a continuación se muestra la página web.

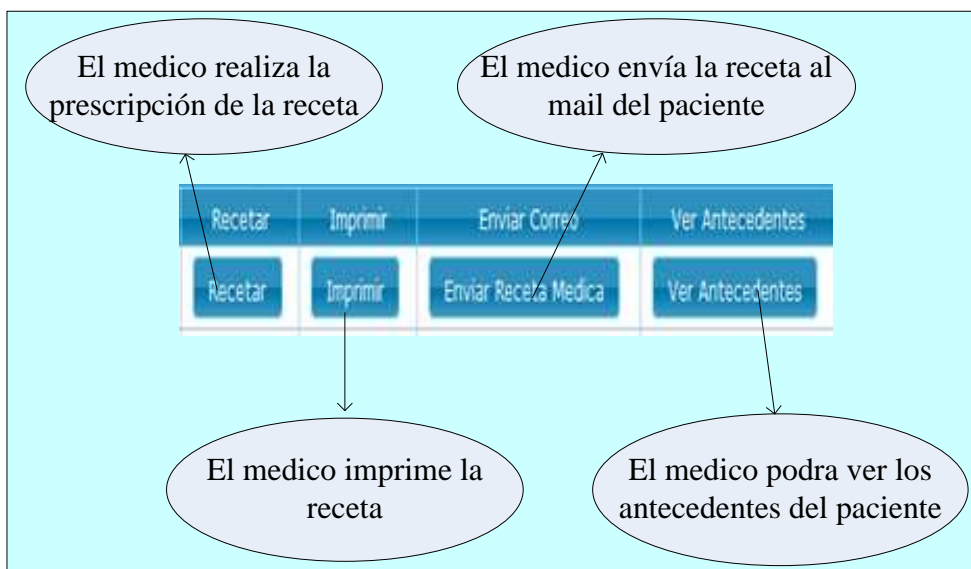
Tendremos en una lista los pacientes que hicieron una consulta al doctor, las cuales van a recibir una prescripción médica, también se enviará la receta electrónica al correo del paciente.

Paciente	Consultas Medicas	Indicaciones Medicas	Preescripciones Medicas	Recetar	Imprimir	Enviar Correo	Ver Antecedentes
Pamela Betancourt	tengo diarrea y vomito	3 veces al dia	jarabe de miel	Recetar	Imprimir	Enviar Receta Medica	Ver Antecedentes
Pamela Betancourt	me duele la cabeza			Recetar	Imprimir	Enviar Receta Medica	Ver Antecedentes

El médico desde esta página web podrá responder a las preguntas que se le realiza a través de chat interno.



Los botones que tenemos en cada uno de los registros de cada paciente son los siguientes:

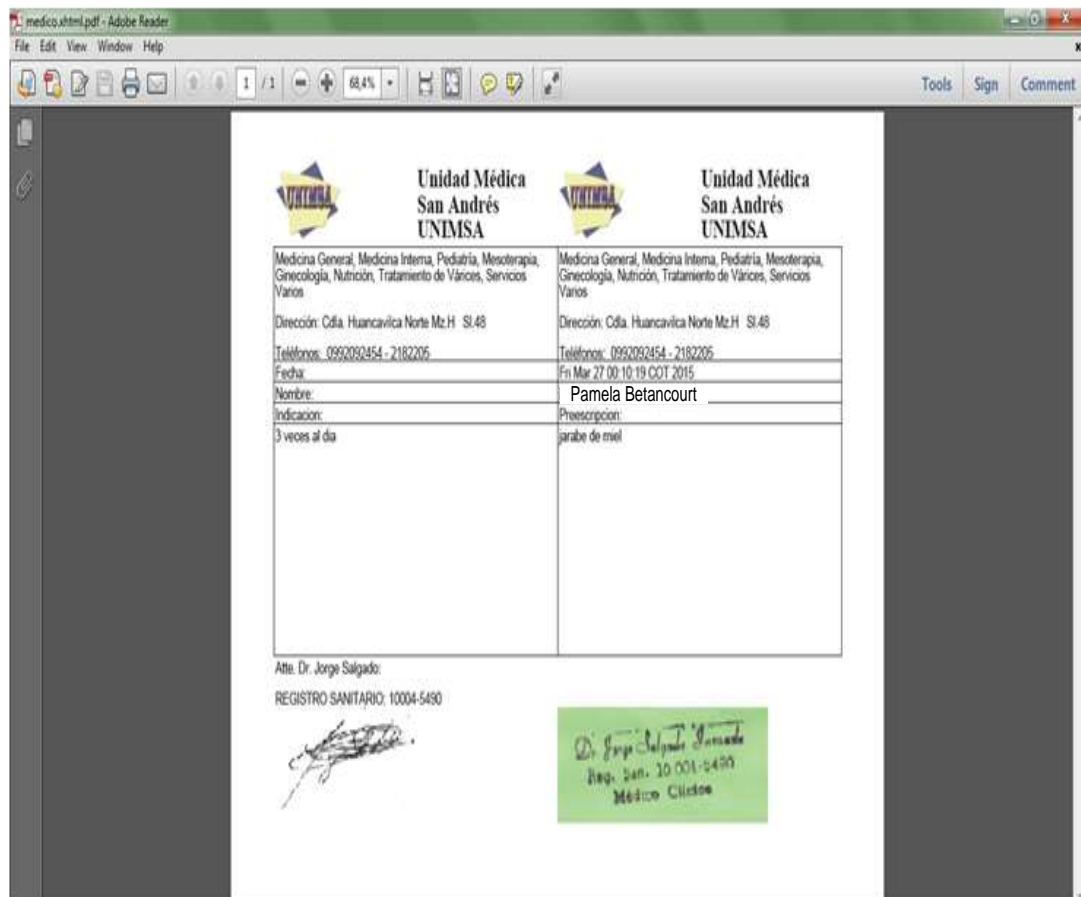


Ver antecedentes:


Al momento de presionar en el botón ver antecedentes se abrirá un cuadro de dialogo en el cual el médico verificará todo los antecedentes que tiene ese paciente con esta información ya puede tener una guía de todo lo que le ha sucedido al paciente en el transcurso de su vida, en esta página el médico no podrá alterar ningún dato.

Al momento de que el médico de clic en el botón recetar se abrirá un formulario en el cual el médico debe ingresar las prescripciones e indicaciones médicas que el paciente necesite.


Al momento de que el médico presione el botón de imprimir aparecerá la receta en formato pdf para poder imprimirla y que el médico pueda tener un respaldo de esta prescripción médica si desea.



Para enviar la receta al correo electrónico del paciente el médico deberá pulsar el botón enviar receta una vez pulsado este botón automáticamente la receta llegará al mail del paciente y se mostrará de la siguiente manera:

Buscar en el correo 

Receta Electronica ↑ ↓ ×

Carpetas 

Bandeja de entrada 4713

Archivo





Correo no deseado 44

Borradores 37

Enviados

Eliminados

[Nueva carpeta](#)

 <p>Unidad Médica San Andrés UNIMSA</p>	 <p>Unidad Médica San Andrés UNIMSA</p>
<p>Medicina General, Medicina Interna, Pediatría, Mesoterapia, Ginecología, Nutrición, Tratamiento de Várices, Servicios Varios Dirección: Cdlá. Huancavilca Norte Mz.H Sl.48 Teléfonos: 0992092454 - 2182205</p>	<p>Medicina General, Medicina Interna, Pediatría, Mesoterapia, Ginecología, Nutrición, Tratamiento de Várices, Servicios Varios Dirección: Cdlá. Huancavilca Norte Mz.H Sl.48 Teléfonos: 0992092454 - 2182205</p>
Fecha:	Fri Mar 27 00:10:46 COT 2015
Nombres:	Pamela Betancourt
Indicacion:	Preescrpcion:
3 veces al dia	jarabe de miel
Att. Jorge Salgado	
REGISTRO SANITARIO: 10004-5490	
	

Leido a2598575.vo.msccnd.net Microsoft Términos Privacidad y cookies Desarrolladores Español

Anexo B Diccionario de Datos

Tabla Instrucción

DICCIONARIO DE DATOS		1			
		Fecha de Diseño: 07/12/2014		Fecha/Actualización: 15/01/2015	
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: instrucción	Tipo de Tabla: Maestra		Base: UNIMSA		
Descripción: Registra los niveles instrucción académica que existen en el país					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDINSTRUC CION	Identificador de la instrucción	PK	VARCHAR (5)	ÚNICO
2	INSTRUCCI ÓN	Nombre de la instrucción	E	VARCHAR (100)	
3	ESTADO	Identificador del estado	E	VARCHAR (1)	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla Grupo Sanguíneo

DICCIONARIO DE DATOS		2			
		Fecha de Diseño: 07/12/2014		Fecha/Actualización: 15/01/2015	
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: grupo sanguíneo	Tipo de Tabla: Maestra		Base: UNIMSA		
Descripción: Registra la clasificación de la sangre					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDGRUPOSA NGUINEO	Identificador del grupo sanguíneo	PK	VARCHAR (5)	ÚNICO
2	GRUPOSAN GUINEO	Nombre del grupo sanguíneo	E	VARCHAR (100)	
3	ESTADO	Identificador del estado	E	VARCHAR (1)	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla Profesiones

DICCIONARIO DE DATOS		3			
		Fecha de Diseño: 07/12/2014		Fecha/Actualización: 15/01/2015	
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: profesiones	Tipo de Tabla: Maestra		Base: UNIMSA		
Descripción: Registra los tipos de profesiones que existen en el país					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDPROFESION	Identificador de la profesión	PK	VARCHAR (5)	ÚNICO
2	PROFESION	Nombre de la profesión	E	VARCHAR (100)	
3	ESTADO	Identificador del estado	E	VARCHAR (1)	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla Estado Civil

DICCIONARIO DE DATOS		4			
		Fecha de Diseño: 07/12/2014		Fecha/Actualización: 15/01/2015	
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: estadocivil	Tipo de Tabla: Maestra		Base: UNIMSA		
Descripción: Registra los tipos de estado civil de un ciudadano ecuatoriano					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDESTADOCIVIL	Identificador del estado civil	PK	VARCHAR (5)	ÚNICO
2	ESTADOCIVIL	Nombre del estado civil	E	VARCHAR (100)	
3	ESTADO	Identificador del estado	E	VARCHAR (1)	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla Receta

DICCIONARIO DE DATOS		5			
		Fecha de Diseño: 07/12/2014	Fecha/Actualización: 15/01/2015		
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: receta	Tipo de Tabla: Detalle		Base: UNIMSA		
Descripción: Registra la receta médica de un paciente					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDRECETA	Identificador de la receta	PK	INT	ÚNICO
2	PRESCRIPCION	Prescripción medica	E	VARCHAR (1000)	OPCIONAL
3	INDICACIONES	Indicación medica	E	VARCHAR (1000)	OPCIONAL
4	IDMEDICO	Identificador del médico	FK	INT	
5	IDPACIENTE	Identificador del Paciente	FK	INT	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla Médico

DICCIONARIO DE DATOS		6			
		Fecha de Diseño: 07/12/2014		Fecha/Actualización: 15/01/2015	
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: medico	Tipo de Tabla: Maestra		Base: UNIMSA		
Descripción: Registra los médicos encargados del dispensario UNIMSA					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDMEDICO	Identificador del médico	PK	INT	ÚNICO
2	NOMBRES	Nombres del médico	E	VARCHAR (100)	
3	ESPECIALIZACION	Especialización del médico	E	VARCHAR (100)	
4	ESTADO	Identificador del estado	E	VARCHAR (1)	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla Paciente

DICCIONARIO DE DATOS		7			
		Fecha de Diseño: 07/12/2014		Fecha/Actualización: 15/01/2015	
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: paciente	Tipo de Tabla: Detalle		Base: UNIMSA		
Descripción: Registra los pacientes que necesitan utilizar el portal					
DESCRIPCIÓN DEL REGISTRO					

No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDPACIENTE	Identificador del paciente	PK	INT	ÚNICO
2	NOMBRES	Nombres del paciente	E	VARCHAR (100)	
3	FECHAREGISTRO	Fecha del registro	E	DATE	OPCIONAL
4	PROFESION	Identificador de la profesión	FK	VARCHAR (5)	
5	INSTRUCCIÓN	Identificador de la instrucción	FK	VARCHAR (5)	
6	OCUPACION ACTUAL	Trabajo actual del paciente	E	VARCHAR (100)	OPCIONAL
7	OCUPACION ANTERIOR	Trabajo anterior del paciente	E	VARCHAR (100)	OPCIONAL
8	ESTADOCIVIL	Identificador del estado civil	E	VARCHAR (5)	
9	MOTIVOCONSULTA	Motivo de usar al portal	E	VARCHAR (500)	OPCIONAL
10	TELEFONO OFICINA	Numero teléfono del trabajo del paciente	E	VARCHAR (45)	OPCIONAL
11	TELEFONO CASA	Numero teléfono de casa del paciente	E	VARCHAR (45)	OPCIONAL
12	TELEFONO CELULAR	Numero celular del paciente	E	VARCHAR (45)	OPCIONAL
13	GRUPOSANGUINEO	Identificador del grupo sanguíneo	FK	VARCHAR (5)	
14	LUGARRESIDENCIA	Lugar donde vive el paciente	E	VARCHAR (100)	
15	FECHANACIMIENTO	Fecha nacimiento	E	DATE	

	MIENTO	del paciente			
16	LUGARNACI MIENTO	Lugar de nacimiento del paciente	E	VARCHAR (100)	
17	CLAVE	Clave que ingresa el paciente	E	VARCHAR (45)	
18	USUARIO	Usuario que ingresa el paciente	E	VARCHAR (45)	
19	EMAIL	Correo electrónico del paciente	E	VARCHAR (45)	
20	EDAD	Edad del paciente	E	INT	
21	APELLIDOS	Apellidos del paciente	E	VARCHAR (100)	
22	ESTADO	Estado del paciente	E	VARCHAR (1)	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla Antecedentes

DICCIONARIO DE DATOS		8			
		Fecha de Diseño: 07/12/2014		Fecha/Actualización: 15/01/2015	
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: Antecedentes	Tipo de Tabla: Detalle		Base: UNIMSA		
Descripción: Registra los Antecedentes que tienen los pacientes que se han registrado en el portal web					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN

1	IDANTECEDENTE	Identificador del antecedente del paciente	PK	INT	ÚNICO
2	EPILEPSIA	Sufre epilepsia	E	BIT	OPCIONAL
3	IDPACIENTE	Identificador del paciente	E	INT	
4	ALTURA	Altura del paciente	FK	DOUBLE	
5	PESO	Peso del paciente	FK	DOUBLE	
6	DEFECATORIO	El paciente es defecatorio	E	BIT	OPCIONAL
7	MICCIONAL	El paciente es miccional	E	BIT	OPCIONAL
8	ALIMENTARIO	El paciente es alimentario	E	BIT	OPCIONAL
9	DEPORTES	El paciente es deportista	E	BIT	OPCIONAL
10	DROGAS	El paciente consume droga	E	BIT	OPCIONAL
11	ALCOHOL	El paciente consume alcohol	E	BIT	OPCIONAL
12	TABACO	El paciente fuma	E	BIT	OPCIONAL
13	OPERACIONES QUIRURGICAS	El paciente tiene operaciones quirúrgicas	FK	LONGTEXT	OPCIONAL
14	ACCIDENTES DE TRABAJO	El paciente ha tenido accidentes de trabajos	E	LONGTEXT	OPCIONAL
15	ENFERMEDADES DE LA INFANCIA	Enfermedades de la infancia del paciente	E	LONGTEXT	OPCIONAL

16	OTRAS INMUNIZACIONES	Nombrar las inmunizaciones del paciente	E	VARCHAR (200)	OPCIONAL
17	HB	Hepatitis B	E	VARCHAR (200)	OPCIONAL
18	HAI	Hepatitis	E	VARCHAR (200)	OPCIONAL
19	TIFOIDEA	El paciente ha tenido tifoidea	E	VARCHAR (200)	OPCIONAL
20	TETANO	El paciente ha tenido tétano	E	VARCHAR (200)	OPCIONAL
21	OTROS	Otras enfermedades	E	VARCHAR (200)	OPCIONAL
22	HA	Hepatitis A	E	BIT	OPCIONAL
23	DOLLUMBAR	El paciente sufre de dolores lumbares	E	BIT	OPCIONAL
24	DIABETE	El paciente sufre de diabetes	E	BIT	OPCIONAL
25	FOBIAS	El paciente sufre de fobias	E	BIT	OPCIONAL
26	ASMA	El paciente sufre de asma	E	BIT	OPCIONAL
27	ALERGIA	El paciente sufre de alergia	E	BIT	OPCIONAL
28	VENEREAS	El paciente sufre de alguna enfermedad venérea	E	BIT	OPCIONAL
29	SILICOSIS	El paciente tiene silicosis	E	BIT	OPCIONAL
30	CA	Tumores	E	BIT	OPCIONAL

31	TUBERCULOSIS	El paciente sufre de tuberculosis	E	BIT	OPCIONAL
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Tabla histórica

DICCIONARIO DE DATOS		9			
		Fecha de Diseño: 07/12/2014	Fecha/Actualización: 15/01/2015		
Para: Universidad Politécnica Salesiana	UNIMSA		Autor: Pamela Betancourt Yépez		
Tabla: historialreceta	Tipo de Tabla: Detalle		Base: UNIMSA		
Descripción: Registra el historial de recetas enviadas					
DESCRIPCIÓN DEL REGISTRO					
No.	CAMPO	DESCRIPCIÓN	TIPO	FORMATO	VALIDACIÓN
1	IDHISRECETA	Identificador de la receta	PK	INT	ÚNICO
2	PRESCRIPCION	Prescripción medica	E	VARCHAR (1000)	OPCIONAL
3	INDICACIONES	Indicación medica	E	VARCHAR (1000)	OPCIONAL
4	IDMEDICO	Identificador del médico	FK	INT	
5	IDPACIENTE	Identificador del Paciente	FK	INT	
6	FECHAREGISTRO	Fecha de registro	E	DATE	
7	ENVIACORREO	Se envió el correo al paciente	E	BIT	
OBSERVACIONES:					
TIPO: PK - Clave Primaria E - Elemento de Dato FK - Clave Foránea					

Anexo C Fotos y Certificado

Doctor Jorge Salgado Director del Dispensario UNIMSA



Capacitando al Doctor con el sistema de prescripción de medicamentos

