



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS,
ELECTRÓNICA E INDUSTRIAL
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS

Tema:

**“APLICACIÓN DISTRIBUIDA BASADO EN LAS ESPECIFICACIONES JEE6
PARA EL CONTROL DEL INVENTARIO DE MEDICINAS EN LA CLÍNICA SAN
BARTOLOMÉ”**

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

SUBLÍNEA DE INVESTIGACIÓN: Aplicación Distribuida.

AUTOR: Juan José Santamaría López

PROFESOR REVISOR: Ing. Clay Aldás

Ambato – Ecuador

Julio – 2014

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de investigación sobre el tema: **“APLICACIÓN DISTRIBUIDA BASADO EN LAS ESPECIFICACIONES JEE6 PARA EL CONTROL DEL INVENTARIO DE MEDICINAS EN LA CLÍNICA SAN BARTOLOMÉ”**, del señor Juan José Santamaría López, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el Art. 16 del Capítulo II, del Reglamento de Graduación para obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

Ambato, Julio 2 de 2014

EL TUTOR

Ing. Clay Aldás

AUTORÍA

El presente trabajo de investigación titulado: “**APLICACIÓN DISTRIBUIDA BASADO EN LAS ESPECIFICACIONES JEE6 PARA EL CONTROL DEL INVENTARIO DE MEDICINAS EN LA CLÍNICA SAN BARTOLOMÉ**”, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, Julio 2 de 2014

Juan José Santamaría López

CI: 1804166534

APROBACIÓN DEL TRIBUNAL DE GRADO

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Renato Urvina e Ing. Galo López, revisó y aprobó el Informe Final del trabajo de graduación titulado “**APLICACIÓN DISTRIBUIDA BASADO EN LAS ESPECIFICACIONES JEE6 PARA EL CONTROL DEL INVENTARIO DE MEDICINAS EN LA CLÍNICA SAN BARTOLOMÉ**”, presentado por el señor Juan José Santamaría López, de acuerdo al Art. 57 del Reglamento de Graduación para obtener el Título Terminal del Tercer Nivel de la Universidad Técnica de Ambato.

Ambato, Julio 2 de 2014

Ing. Vicente Morales L., Mg.

PRESIDENTE DEL TRIBUNAL

Ing. Renato Urvina, Mg.

DOCENTE CALIFICADOR

Ing. Galo López, Mg.

DOCENTE CALIFICADOR

DEDICATORIA

Dedico esta tesis principalmente a Dios, por permitirme el haber llegado hasta este momento tan importante de mi formación profesional. A mi madre, por ser el pilar más importante en mi vida y por demostrarme siempre su cariño y apoyo incondicional sin importar las circunstancias que estemos atravesando. A mi padre que de una u otra manera me demostró que en la vida todo se puede lograr con esfuerzo y dedicación, enfrentando todas las situaciones adversas que se nos presenten. Sin el gran apoyo que los dos me brindaron no hubiese logrado esta meta.

Juan José Santamaría López

AGRADECIMIENTO

Agradezco a Dios por protegerme durante todo mi camino y darme fuerzas para superar obstáculos y dificultades a lo largo de toda esta etapa estudiantil.

A mis tíos Pablo y Cipriano, por su apoyo incondicional y por demostrarme toda la confianza que tienen en mí.

A todos mis compañeros de trabajo en la empresa NDeveloper por todas las enseñanzas que me brindaron.

Al Ing. Clay Aldás, tutor de tesis, por su valiosa ayuda y asesoramiento a la realización de la misma.

En fin agradezco a todas las personas que ayudaron directa e indirectamente en la realización de este proyecto.

Juan José Santamaría López

Índice de Contenidos

CAPÍTULO 1 : EL PROBLEMA	1
1.1. Tema de Investigación.....	1
1.2. Planteamiento del Problema.....	1
1.3. Delimitación	2
1.4. Justificación.....	3
1.5. Objetivos	4
1.5.1. Objetivo General	4
1.5.2. Objetivos Específicos	4
CAPÍTULO 2 : MARCO TEÓRICO.....	5
2.1. Marco Teórico	5
2.1.1. Control de Inventario.....	5
2.1.2. Aplicación Distribuida.....	6
2.1.3. Fundamentación	10
CAPÍTULO 3 : METODOLOGÍA	16
3.1. Modalidad de la Investigación	16
3.2. Recolección de la Información.....	16
3.3. Procesamiento y Análisis de la Información.....	17
3.4. Desarrollo del Proyecto.....	18
CAPÍTULO 4 : DESARROLLO DE LA PROPUESTA	22
4.1. Generación del Product Backlog.....	23
4.2. Sprint Planning.....	29
4.3. Sprints	29
4.3.1. Primer Sprint	29
4.3.2. Segundo Sprint	37
4.3.3. Tercer Sprint	41
4.3.4. Cuarto Sprint	44
4.4. Versión final de la Aplicación	48

4.5. Modelo de la base de datos	49
4.6. Implantación de la aplicación.....	49
4.6.1. Requerimientos de Software.....	49
4.6.2. Configuración del servidor de aplicaciones	50
4.6.3. Cronograma de Implantación	50
4.6.4. Cronograma de Capacitación	51
CAPÍTULO 5 : CONCLUSIONES Y RECOMENDACIONES.....	61
5.1 Conclusiones	61
5.2. Recomendaciones.....	62
Bibliografía	64
Anexos	66
Anexo 1: Manual de Usuario	66
Anexo 2: Manual Técnico	101

Índice de Tablas

Tabla 4-1 Historia de Usuario de Ingreso de Laboratorio.....	23
Tabla 4-2 Historia de Usuario de Ingreso Tipos de Usuario.....	24
Tabla 4-3 Historia de Usuario de Ingreso de Usuarios que utilizan la aplicación.	24
Tabla 4-4 Historia de Usuario de Ingreso de Visitadores Médicos.....	25
Tabla 4-5 Historia de Usuario de Ingreso de Tipo Presentación para la medicina.	25
Tabla 4-6 Historia de Usuario de Ingreso de Destinos de la medicina.	26
Tabla 4-7 Historia de Usuario de Ingreso de Medicinas.....	26
Tabla 4-8 Historia de Usuario de Registro de Egresos de Medicina.	27
Tabla 4-9 Historia de Usuario de Registro de Ingresos de Medicina.	27
Tabla 4-10 Historia de Usuario de Menú de la aplicación.....	28
Tabla 4-11 Historia de Usuario de Reporte de medicinas.....	28
Tabla 4-12 Cronograma de Implantación	50
Tabla 4-13 Cronograma de Capacitación.....	51
Tabla 4-14 Lista de Atributos de la tabla destino_medicina	54
Tabla 4-15 Lista de Atributos de la tabla egreso_detalle.....	54
Tabla 4-16 Lista de Atributos de la tabla egresos	54
Tabla 4-17 Lista de Atributos de la tabla ingreso_detalle	55
Tabla 4-18 Lista de Atributos de la tabla ingresos	55
Tabla 4-19 Lista de Atributos de la tabla laboratorios	56
Tabla 4-20 Lista de Atributos de la tabla medicinas	56
Tabla 4-21 Lista de Atributos de la tabla menú	57
Tabla 4-22 Lista de Atributos de la tabla tipo_presentacion.....	57
Tabla 4-23 Lista de Atributos de la tabla tipo_usuario	57
Tabla 4-24 Lista de Atributos de la tabla usuarios.....	58
Tabla 4-25 Lista de Atributos de la tabla visitadores.....	59

Índice de Figuras

Fig 3-1 Proceso manual de control de inventario de medicinas	18
Fig 3-2 Proceso de Scrum	21
Fig 4-1 Método para el ingreso de laboratorio	30
Fig 4-2 Método para la modificación de un laboratorio	30
Fig 4-3 Método para guardar (controlador)	31
Fig 4-4 Método para guardar (servicio)	31
Fig 4-5 Método para guardar (servicio de acceso a la base de datos)	31
Fig 4-6 Prueba de unidad método guardar	32
Fig 4-7 Método para eliminar (controlador)	32
Fig 4-8 Método para eliminar (servicio)	32
Fig 4-9 Método para eliminar (servicio de acceso a la BDD)	33
Fig 4-10 Interfaz de manipulación de datos	33
Fig 4-11 Validación de campos requeridos	34
Fig 4-12 Interfaz de Laboratorios	36
Fig 4-13 Interfaz de Tipos de Usuario	37
Fig 4-14 Interfaz de Usuarios	37
Fig 4-15 Interfaz de Visitadores	40
Fig 4-16 Interfaz de Tipos Presentación de Medicinas	40
Fig 4-17 Interfaz de Destino de Medicina	41
Fig 4-18 Interfaz de Medicinas	43
Fig 4-19 Interfaz de Egreso de Medicina 1	43
Fig 4-20 Interfaz de Egreso de Medicina 2	44
Fig 4-21 Interfaz de Ingreso de Medicina 1	46
Fig 4-22 Interfaz Ingreso de Medicina 2	46
Fig 4-23 Menú de la aplicación 1	47
Fig 4-24 Menú de la aplicación 2	47
Fig 4-25 Sección de Reportes	47
Fig 4-26 Reporte Inventario General	48
Fig 4-27 Reporte por Medicina	48
Fig 4-28 Reporte Medicinas Caducadas	48
Fig 4-29 Modelo Relacional Base de Datos	52
Fig 4-30 Modelo Entidad Relacion de Base de Datos	53
Fig 4-31 Configuración Datasourse	59
Fig 4-32 Arquitectura de la aplicación distribuida	60

RESUMEN

En la actualidad, el control de inventario de medicinas en las instituciones médicas es muy importante tanto para estas como para el país, por esta razón deben estar en continua evolución con la tecnología y así cumplir con sus procesos de manera eficaz.

Para mejorar el proceso de Control de Inventario de Medicinas de la Clínica San Bartolomé se ha desarrollado una Aplicación Distribuida que permite realizar el control de Inventario de medicinas y así dejar de realizar este proceso manualmente.

Este proyecto contiene toda la información necesaria para mejorar el proceso de Control de Inventario de Medicinas que se maneja en la Institución Médica, mediante la aplicación desarrollada, se ha optimizado tiempos en la verificación de las existencias de medicinas en el inventario.

La aplicación distribuida para el control del inventario de medicinas fue elaborada bajo las especificaciones JEE6 dentro de las cuales consta por ejemplo el uso del servidor de aplicaciones JBOSS para ejecutar el sistema, este es un servidor de aplicaciones muy conocido en los ambientes de desarrollo Java, es estable y de fácil configuración.

INTRODUCCIÓN

El desarrollo del informe final del presente trabajo de investigación se encuentra dividido en capítulos, los mismos que facilitan la comprensión del contenido del mismo.

CAPÍTULO 1

Denominado “EL PROBLEMA”, se identifica el problema a investigar, la justificación respectiva de la investigación y el planteamiento de los objetivos a obtener tras la culminación del trabajo de investigación.

CAPÍTULO 2

Contiene el “MARCO TEÓRICO”, en donde se establece el conjunto de conocimientos en los cuales se sustenta la investigación, además de presentar investigaciones previas que sirven de soporte a la investigación y se establece la propuesta de solución del problema.

CAPÍTULO 3

Comprende la “METODOLOGÍA”, en la cual se especifica la metodología de investigación a utilizar, el proceso de recolección de la información, el procesamiento y análisis de la información recabada y define las etapas para el desarrollo del proyecto.

CAPÍTULO 4

Consta del “DESARROLLO DE LA PROPUESTA”, que detalla la metodología seleccionada para el desarrollo de la Aplicación.

CAPÍTULO 5

Este capítulo contiene las “CONCLUSIONES Y RECOMENDACIONES” a las cuales se llegan al finalizar el proyecto.

Y por último se encuentran los anexos correspondientes al trabajo de investigación.

CAPÍTULO 1 : EL PROBLEMA

1.1. Tema de Investigación

APLICACIÓN DISTRIBUIDA BASADO EN LAS ESPECIFICACIONES JEE6 PARA EL CONTROL DEL INVENTARIO DE MEDICINAS EN LA CLÍNICA SAN BARTOLOMÉ

1.2. Planteamiento del Problema

A nivel mundial en toda organización o empresa su base consta de la compra y venta de productos y/o servicios, por esta razón nace la importancia de la administración de procesos tales como el control de inventario o la facturación. Todos estos procesos ayudan a las empresas a nivel mundial a conocer la situación económica de la misma dentro de un periodo contable.

Es por eso que en todas las empresas de salud pública y privada del Ecuador deberán tener información confiable, real y oportuna la cual ayudará en la toma de decisiones, es así que en cada institución de salud debe existir un control efectivo e inmediato de los procesos que intervienen en un control de inventario de medicinas, con esto evitar errores que pueden provocar pérdidas para la institución.

El control del inventario de medicinas de las clínicas de la provincia Tungurahua requiere de una serie de procesos cíclicos para obtener la información exacta de las existencias, para luego ser analizadas para la toma de decisiones, además al no saber con exactitud la cantidad de medicina que tiene la clínica, se facilita la pérdida de medicina. Las clínicas de la provincia al no contar con un control de inventario organizado y estructurado han tenido que enfrentarse a problemas como la pérdida de información lo cual perjudica al control de todas la actividades que se realizan en estas instituciones.

Dentro de este contexto la Clínica San Bartolomé no es la excepción ya que cuenta con un débil control de inventario de medicinas debido a que los procesos de conteo de existencias y registro de medicinas se los realiza manualmente provocando pérdida de tiempo de las personas que realizan estos procesos.

1.3. Delimitación

Área:

Software

Línea de investigación:

Desarrollo de Software

Sublínea de Investigación:

Aplicaciones Distribuidas

Delimitación Espacial:

La presente investigación de llevara a cabo en la parroquia San Bartolomé de Pinllo en la Clínica San Bartolomé.

Delimitación Temporal:

Seis meses a partir de la aprobación del proyecto.

1.4. Justificación

La base de toda institución comercial es la compra y venta de bienes o servicios; de aquí la importancia del control del inventario por parte de la misma. Este manejo contable permitirá a la empresa mantener el control de existencias de medicinas oportunamente, así como también conocer al final del periodo contable un estado confiable de la situación económica de la empresa.

La investigación propuesta busca el estudio y análisis de la teoría y conceptos básicos del control de inventarios, contribuir con la clínica San Bartolomé.

La presente investigación se la realiza por la deficiente administración del inventario de medicinas en una clínica ya que el proceso se lo realiza manualmente, siendo esto deficiente.

El uso de sistemas automatizados para llevar el control del inventario de medicinas ayudará a ser lo más exactos posibles en la entrega de informes para conocer el estado del inventario de medicina.

La utilización de sistemas automatizados ha sido de mucha ayuda en las clínicas ya que se evita caer en errores de cálculos y procesos dando resultados óptimos y adecuados a las personas a las que está dirigido el sistema.

Es factible ya que se me permitirá obtener toda la información necesaria para la realización del sistema.

1.5. Objetivos

1.5.1. Objetivo General

- Desarrollar una Aplicación Distribuida basada en las Especificaciones JEE6 para el control del inventario de medicinas en la Clínica San Bartolomé.

1.5.2. Objetivos Específicos

- Establecer los procesos para el control de inventario de medicinas.
- Analizar e identificar los requerimientos de la Aplicación Distribuida basada en las Especificaciones JEE6 en la Clínica San Bartolomé.
- Diseñar y establecer la arquitectura y el funcionamiento de la Aplicación Distribuida.
- Implementar la Aplicación Distribuida basada en las especificaciones JEE6.

CAPÍTULO 2 : MARCO TEÓRICO

2.1. Marco Teórico

2.1.1. Control de Inventario

“El control efectivo de los inventarios consiste en el equilibrio entre lo que se tiene en inventario y las necesidades de producción y ventas. Si se presentan faltantes para la producción, ésta se interrumpe. Si se tienen faltantes en el inventario de producto terminado, no se puede cumplir con los pedidos de los clientes. Por lo tanto, el control de inventarios involucra la planeación de la producción, las compras y el pronóstico de las ventas. Con el fin de evitar los faltantes en los inventarios, y de tener una óptima rotación de los mismos, se han desarrollado técnicas, algunas de las cuales ya han sido mencionadas con respecto a la función de compras.” [1]

“El control de inventario se refiere a obtener un equilibrio entre dos objetivos opuestos: 1) minimizar el costo de mantener un inventario y 2) maximizar el servicio a los clientes. Los costos de inventario incluyen los costos de inversión, de almacenamiento y de las obsolescencias o daños posibles. Con frecuencia el costo de inversión es el factor dominante; un caso común es cuando la compañía invierte dinero prestado a cierta tasa de interés en materiales que todavía no se han entregado al cliente. Todos estos costos se denominan costos para mantener un inventario. La empresa puede minimizar esos costos si mantiene los inventarios en cero, sin embargo, es probable que esto afecte el servicio a los clientes y decidan hacer negocios en otra parte. Lo anterior representa un costo, denominado costo de materiales agotados (stock-out cost). Una compañía prudente

pretende minimizar el costo de material agotado y ofrecer un alto nivel de servicios al cliente. Este último concepto implica tanto los clientes externos (los que generalmente se asocian con esta palabra) y los clientes internos, que son los departamentos operativos, de ensamble final y otras unidades en la organización que dependen de la disponibilidad de materiales y panes.” [2]

2.1.2. Aplicación Distribuida

“Una aplicación distribuida proporciona una funcionalidad que ha sido dividida en un conjunto de unidades funcionales autónomas formados por datos y operaciones para manipular estos datos. Las operaciones representan el comportamiento en forma de servicios que ofrecen al mundo exterior, mientras que los datos representan su estado interno. Las unidades funcionales o componentes asignan a diferentes máquinas de un sistema distribuido e interactúan entre ellos mediante interfaces claramente definidas y documentadas. Estas interfaces son la pieza clave en las aplicaciones distribuidas que permiten interactuar entre componentes solamente mostrando los servicios que ofrecen. Las podemos ver como las firmas de los métodos típicos de una aplicación cualquiera, donde también se incluye los tipos de datos que aceptan de entrada y el tipo de resultado que devuelven.

En la computación distribuida, las aplicaciones distribuidas pueden estar formadas por un conjunto de componentes no homogéneos, algunos de ellos de nueva construcción debido a la natural evolución del sistema, y otros más antiguos pertenecientes a sistemas existentes. Los componentes pueden encontrarse escritos en diferentes lenguajes y soportados por diferentes sistemas operativos y distinto hardware. Aun así, estos componentes deben interactuar entre ellos de modo que cada uno satisfaga las peticiones de servicio solicitadas tanto por los otros componentes como por el usuario final.” [3]

“Intervienen varias partes, que no tienen porqué estar localizadas físicamente en el mismo lugar, se comparten recursos, interfaces abiertos no propietarios, son escalables.

Para el desarrollo de aplicaciones distribuidas existen distintas tecnologías. Hasta el momento las más utilizadas han sido:

- Cobra: (Common Object Request Broker Architecture), el planteamiento de esta tecnología es facilitar la invocación de procedimientos en “x” lenguaje a partir de otro lenguaje distinto. Se basa en la definición de interfaces mediante IDL (Interface Definition Language).
- DCOM: (Distributed Component Object Model), es un programa binario de Microsoft que permite que aplicaciones desarrolladas en distintos lenguajes y situadas en distintas máquinas puedan compartir objetos, es decir, con DCOM podremos desde una aplicación acceder métodos y propiedades de objetos de otras aplicaciones.
- EJB: (Enterprise Java Beans), Es un componente que permite agrupar funcionalidades de una aplicación y requiere encontrarse en un ambiente de ejecución. Mediante un EJB container se ofrecen los servicios y a través de un EJB es posible desarrollar los componentes principales. Su diseño es demasiado complejo.
- RMI: (Remote Method Invocation), mecanismo de invocación/ejecución de procedimientos remotos de computadoras distribuidas. Una de las ventajas al diseñar un procedimiento con RMI es la interoperabilidad, ya que RMI forma parte del JDK de Java, por lo que cualquier plataforma que tenga acceso a un JDK también lo tendrá a estos procedimientos. Es mucho más sencillo que los anteriores.” [4]

Ventajas de una Aplicación Distribuida:

- “Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no autorizado no pueda dañar el sistema. Esta centralización también facilita la tarea de poner al día datos u otros recursos (mejor que en las redes P2P).

- Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado. Cualquier elemento puede ser aumentado (o mejorado) en cualquier momento, o se pueden añadir nuevos nodos a la red (clientes y/o servidores).
- Fácil mantenimiento: al estar distribuidas las funciones y responsabilidades entre varios ordenadores independientes, es posible reemplazar, reparar, actualizar, o incluso trasladar un servidor, mientras que sus clientes no se verán afectados por ese cambio (o se afectarán mínimamente). Esta independencia de los cambios también se conoce como encapsulación.
- Existen tecnologías, suficientemente desarrolladas, diseñadas para el paradigma de C/S que aseguran la seguridad en las transacciones, la amigabilidad del interfaz, y la facilidad de empleo.”[5]

Características de una Aplicación Distribuida

- “Autonomía Local:
 - Los sitios de una Aplicación Distribuida deben ser autónomos.
 - Ningún sitio X deberá depender de un sitio Y para su buen funcionamiento.
 - Existencia de un propietario y administración local de los datos.
- No dependencia de un sitio central:
 - No debe haber dependencia de un sitio central “maestro” para obtener un servicio.
- Operación continua:
 - Idealmente nunca debería haber necesidad de apagar a propósito el sistema, por ejemplo, para añadir un nuevo sitio o instalar una nueva versión mejorada.
- Independencia con respecto a la localización:

- No debe ser necesario que los usuarios sepan donde están almacenados físicamente los datos.
- Simplifica los programas de los usuarios.
- Permite modificar la distribución de los datos dentro de la red.
- Independencia respecto a la fragmentación.
 - Los usuarios deberán poder comportarse como si los datos no estuvieran fragmentados en realidad.
- Procesamiento distribuido de consultas:
 - En una consulta distribuida, habrá muchas maneras de trasladar los datos en la red para satisfacer la solicitud.
 - Importancia crucial de la optimización.
- Manejo distribuido de transacciones:
 - Control de recuperación: el sistema debe asegurar que cada transacción sea automática (todo o nada).
 - Control de concurrencia: basada en el bloqueo.
- Independencia:
 - Respecto al equipo: máquinas diferentes.
 - Respecto al sistema operativo.
 - Respecto a la red.
 - Respecto al DBMS: comunicación mediante SQL.” [6]

Arquitectura de una Aplicación Distribuida

“Describe la estructura y la organización de los componentes del software, sus propiedades y la conexión entre ellos para formar el sistema; la cantidad y la granularidad de comunicación que se necesita para la interacción y los protocolos de interfaz usada por la comunicación.

En una aplicación distribuida en n-capas los diferentes elementos que integran la aplicación se agrupan de forma lógica según la funcionalidad que reciben o suministran al o desde el resto de los elementos. Así, algunos elementos se limitarán a recibir peticiones de datos mientras que otros interactuarán con el usuario y su función será principalmente la de solicitar a otros elementos la información que el usuario precisa.

Una vez agrupada la funcionalidad en capas lógicas es fácil relacionar unas con otras. El usuario interactuará con la capa de presentación, solicitando datos o desencadenando acciones. Las solicitudes serán atendidas por la capa de negocios, que se encargará de su gestión o de la traducción necesaria para que la capa de servidor realice la tarea solicitada. La capa de servidor debe proporcionar datos los cuales se devolverán a la capa de negocios, la cual los gestionará o transmitirá a la capa de presentación.” [7]

2.1.3. Fundamentación

Capa de datos

Para el desarrollo de la capa de datos se utilizó la siguiente herramienta:

PostgreSQL

“PostgreSQL es un potente sistema de base de datos objeto-relacional de código abierto. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de fiabilidad, integridad de datos y la corrección. Funciona en todos los principales sistemas operativos, incluyendo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows. Es totalmente compatible con ACID, tiene soporte completo para claves foráneas, uniones, vistas, disparadores y procedimientos

almacenados (en varios idiomas). Incluye más de SQL: 2008 tipos de datos, incluyendo INTEGER, numéricos, booleanos, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. También es compatible con el almacenamiento de grandes objetos binarios, como imágenes, sonidos o vídeo. Cuenta con interfaces de programación nativas para C / C ++, Java, . Net, Perl, Python, Ruby, Tcl, ODBC, entre otros.” [8]

Capa de negocio

Para el desarrollo de la capa de negocio se utilizó el lenguaje de programación Java complementado con las especificaciones JEE6.

Java

“Java es un lenguaje de programación y la primera plataforma informática creada por Sun Microsystems en 1995. Es la tecnología subyacente que permite el uso de programas punteros, como herramientas, juegos y aplicaciones de negocios. Java se ejecuta en más de 850 millones de ordenadores personales de todo el mundo y en miles de millones de dispositivos, como dispositivos móviles y aparatos de televisión.

Java ha sido probado, ajustado, ampliado y probado por toda una comunidad de desarrolladores, arquitectos de aplicaciones y entusiastas de Java. Java está diseñado para permitir el desarrollo de aplicaciones portátiles de elevado rendimiento para el más amplio rango de plataformas informáticas posible. Al poner a disposición de todo el mundo aplicaciones en entornos heterogéneos, las empresas pueden proporcionar más servicios y mejorar la productividad, las comunicaciones y colaboración del usuario final y reducir drásticamente el costo de propiedad tanto para aplicaciones de usuario como de empresa. Java se ha convertido en un valor impagable para los desarrolladores, ya que les permite:

- Escribir software en una plataforma y ejecutarla virtualmente en otra
- Crear programas que se puedan ejecutar en un explorador y acceder a servicios Web disponibles

- Desarrollar aplicaciones de servidor para foros en línea, almacenes, encuestas, procesamiento de formularios HTML y mucho más
- Combinar aplicaciones o servicios que utilizan el lenguaje Java para crear aplicaciones o servicios con un gran nivel de personalización
- Escribir aplicaciones potentes y eficaces para teléfonos móviles, procesadores remotos, microcontroladores, módulos inalámbricos, sensores, gateways, productos de consumo y prácticamente cualquier otro dispositivo electrónico.” [9]

Especificaciones JEE6

“Java EE 6 introduce el concepto de perfil, lo que representa una configuración de la plataforma adecuada para una determinada clase de aplicaciones. El perfil web ofrece una pila completa, con tecnologías que aborden la presentación y gestión estatal (JavaServer Faces, JavaServer Pages), la funcionalidad de contenedor Web subyacente (Servlet), la lógica de negocio (Enterprise JavaBeans Lite), transacciones (Java Transaction API), persistencia (Java Persistence API) y más.” [10]

Capa del servidor

Para la capa del servidor se utilizaron las siguientes aplicaciones:

Servidor de Aplicaciones JBoss

“JBoss es un servidor de aplicaciones Java EE de código abierto implementado en Java puro. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo para el que esté disponible la máquina virtual de Java.

JBoss AS es el primer servidor de aplicaciones de código abierto, preparado para la producción y certificado J2EE 1.4, disponible en el mercado, ofreciendo una plataforma de alto rendimiento para aplicaciones de e-business. Combinando una arquitectura orientada a servicios SOA, con una licencia GNU de código abierto, JBoss AS puede ser descargado, utilizado, incrustado y distribuido sin restricciones por la licencia.

Las características destacadas de JBoss incluyen:

- Producto de licencia de código abierto sin coste adicional.
- Cumple los estándares.
- Confiable a nivel de empresa
- Incrustable, orientado a arquitectura de servicios.
- Flexibilidad consistente
- Servicios del middleware para cualquier objeto de Java.
- Soporte completo para JMX.” [10]

Apache Maven

“Apache Maven es una herramienta de gestión y construcción de proyectos de software Java. Maven utiliza un Project Object Model (POM) para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externos, y el orden de construcción de los elementos. Viene con objetivos predefinidos para realizar ciertas tareas claramente definidas, como la compilación del código y su empaquetado.

Una característica clave de Maven es que está listo para usar en red. El motor incluido en su núcleo puede dinámicamente descargar plugins de un repositorio, el mismo repositorio que provee acceso a muchas versiones de diferentes proyectos Open Source en Java, de Apache y otras organizaciones y desarrolladores. Este repositorio y su sucesor reorganizado, el repositorio Maven 2, pugnan por ser el mecanismo de facto de distribución de aplicaciones en Java, pero su adopción ha sido muy lenta. Maven provee soporte no sólo para obtener archivos de su repositorio, sino también para subir artefactos al repositorio al final de la construcción de la aplicación, dejándola al acceso de todos los usuarios. Una caché local de artefactos actúa como la primera fuente para sincronizar la salida de los proyectos a un sistema local.” [11]

Capa de presentación

Para el desarrollo de la capa de presentación se utilizaron las siguientes herramientas:

JasperReport

“JasperReports es el motor de informes de código abierto más popular del mundo. Está escrito completamente en Java y es capaz de utilizar los datos procedentes de cualquier tipo de fuente de datos y producir documentos de píxel perfecto que se pueden ver, imprimir o exportar en una variedad de formatos de documentos incluyendo HTML, PDF, Excel, OpenOffice y Word.” [12].

CSS

“CSS son las siglas de Cascading Style Sheets - Hojas de Estilo en Cascada - que es un lenguaje que describe la presentación de los documentos estructurados en hojas de estilo para diferentes métodos de interpretación, es decir, describe cómo se va a mostrar un documento en pantalla, por impresora, por voz (cuando la información es pronunciada a través de un dispositivo de lectura) o en dispositivos táctiles basados en Braille” [13].

2.2. Propuesta de la solución

Con la implantación de una Aplicación Distribuida para el control del inventario de medicinas en la Clínica San Bartolomé, se resolverá el problema del: Ineficiente proceso manual de control de inventario de medicinas que se maneja en la Clínica San Bartolomé, lo cual provoca pérdida de dinero debido a pérdidas en la existencia de la medicina, además de pérdida de tiempo en el conteo manual de la medicina existente. Con la elaboración de la aplicación distribuida se conseguirá optimizar los procesos utilizados en el control del inventario, además de mejorar el tiempo de ejecución de los mismos, evitando así pérdida de dinero y tiempo, permitiéndole también al usuario la generación de reportes que le servirá para la toma de decisiones.

Esta información la obtuve con entrevistas a la administradora del inventario de la clínica la cual me comentó que en el control diario del inventario que ella lleva se da cuenta de que existen inconsistencias al comparar su registro con la cantidad física en el stock, y que cuando le toca hacer el conteo manual del inventario se demoraba al menos 8 horas en hacerlo lo cual es una pérdida de tiempo.

CAPÍTULO 3 : METODOLOGÍA

3.1. Modalidad de la Investigación

En el desarrollo de la investigación se empleó la investigación de campo ya que ha permitido recolectar datos a través de los individuos que son motivo de estudio, en este caso el personal que labora en la Clínica San Bartolomé.

Se realizó también una investigación bibliográfica para la elaboración del marco teórico, en donde se establece el conjunto de conocimientos en los cuales se sustenta la investigación, además de presentar investigaciones previas que sirven de soporte a la investigación.

3.2. Recolección de la Información

Toda la información que se recolectó sirvió para alcanzar los objetivos establecidos al inicio de la investigación

La información se la recolectó de las personas que trabajan en la clínica San Bartolomé las cuales saben de los procesos que se utilizan en el control del inventario, además obtendremos información del software y hardware que disponemos actualmente, también de la documentación existente en la clínica.

La investigación se basó en la antigüedad, títulos, reglamentación, políticas empresariales, hardware y software que tiene la clínica San Bartolomé.

Esta investigación se la realizó en el periodo de mayo 2012 febrero 2013

La investigación se la realizó en la clínica San Bartolomé que está ubicada en la ciudad de Ambato en la parroquia Pinllo.

Las entrevistas al personal de la clínica se las realizaron en varias ocasiones hasta completar toda la información que necesitamos para terminar la investigación.

Para recolectar toda la información se utilizaron técnicas conocidas como encuestas y entrevistas al personal de la empresa.

Con motivos investigativos por graduación.

3.3. Procesamiento y Análisis de la Información

Una vez recolectada la información se procedió al análisis de los datos obtenidos los cuales fueron de gran importancia para la formulación de la propuesta. Los datos fueron analizados y procesados en relación al problema para poder establecer las respectivas conclusiones asegurando que los datos sean lo más reales posibles.

Como resultado de las entrevistas aplicadas a personal que labora en la Clínica San Bartolomé se ha recabado la siguiente información:

El personal que labora directa o indirectamente en la Clínica San Bartolomé está dividido en grupos, que son: Médicos, Enfermeras, Administradores y Visitadores.

Todos estos grupos intervienen en el inventario de medicinas de la clínica, ya sea agregando o quitando medicinas cumpliendo de acuerdo a los permisos que tiene sobre el inventario.

Una vez realizadas las encuestas y las entrevistas al personal de la Clínica San Bartolomé, se llegó a la conclusión que la Clínica no cuenta con una Aplicación Distribuida para llevar a cabo el proceso de Control de Inventario de medicinas, es decir se lo realiza de forma

manual lo que conlleva a un sinnúmero de contratiempos por tal motivo necesita un Sistema Informático que permita el control del inventario de medicinas.

Proceso manual mediante el cual la Clínica San Bartolomé realiza el control de Inventario de medicinas.

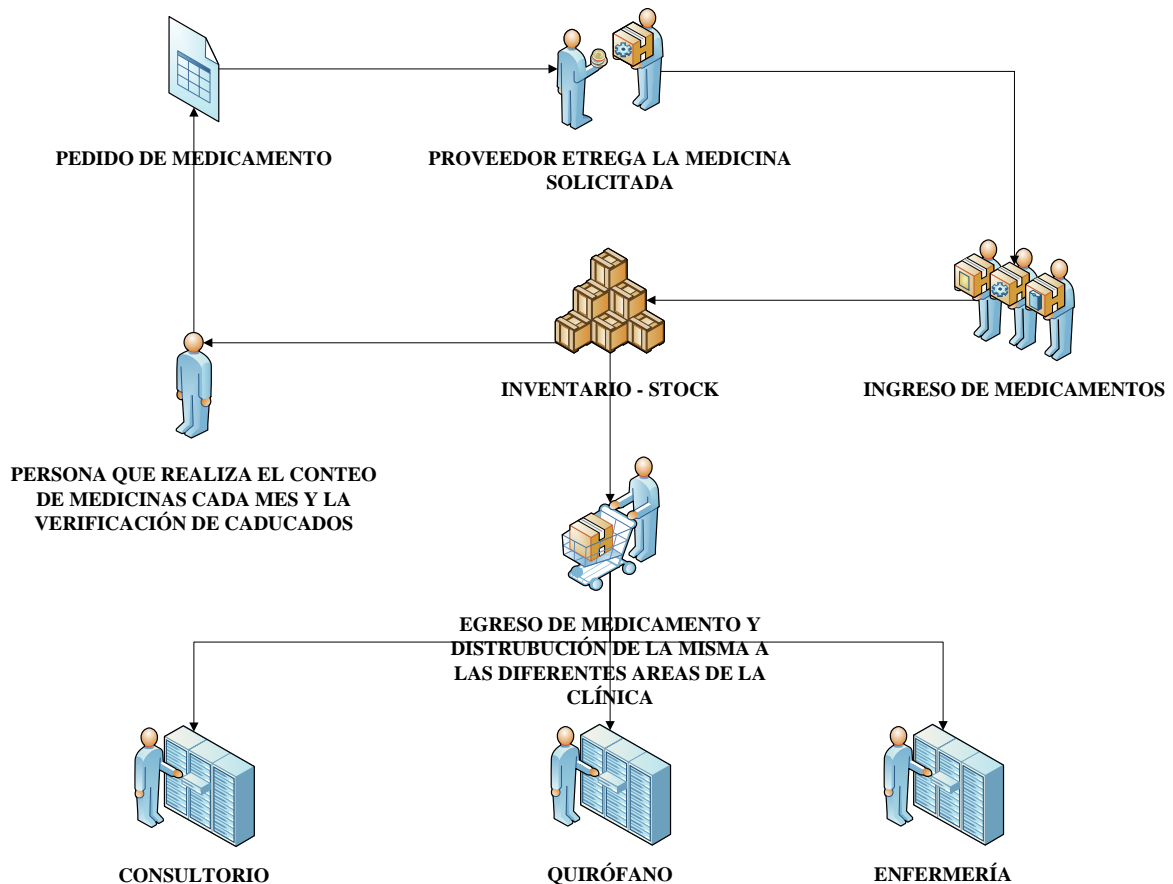


Fig 3-1 Proceso manual de control de inventario de medicinas
Elaborado por: Juan Santamaría

3.4. Desarrollo del Proyecto

A continuación se detallan las Actividades que se realizaron, para cumplir los objetivos que llevaron a la obtención del producto final, teniendo en cuenta que para el desarrollo del proyecto se ha utilizado la metodología de desarrollo ágil: SCRUM.

A continuación se describe brevemente el proceso a seguir en esta metodología para el desarrollo de aplicaciones.

Proceso y Roles de SCRUM

“El proceso

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En cada nuevo Sprint, se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones priorizándose siempre aquellas que aporten mayor valor de negocio. (*Ver Fig 3.2 Proceso de Scrum*)

- **Product Backlog:** Conjunto de requisitos denominados historias de usuario descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.
- **Sprint Planning:** Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint.
- **Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo.
- **Sprint Backlog:** Lista de las tareas necesarias para llevar a cabo las historias del sprint.
- **Daily sprint meeting:** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.

- **Demo y retrospectiva:** Reunión que se celebra al final del sprint y en la que el equipo presenta las historias conseguidas mediante una demostración del producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos.

Roles

En Scrum, el equipo se focaliza en construir software de calidad. La gestión de un proyecto Scrum se centra en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo.

El equipo Scrum está formado por los siguientes roles:

- **Scrum master:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto.
- **Product owner (PO):** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio y el es responsable de la inversión del proyecto (entregar un valor superior al dinero invertido). Traslada la visión del proyecto al equipo, formaliza las prestaciones en **historias** a incorporar en el **Product Backlog** y las reprioriza de forma regular.
- **Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las **historias** a las que se comprometen al inicio de cada sprint.” [15]

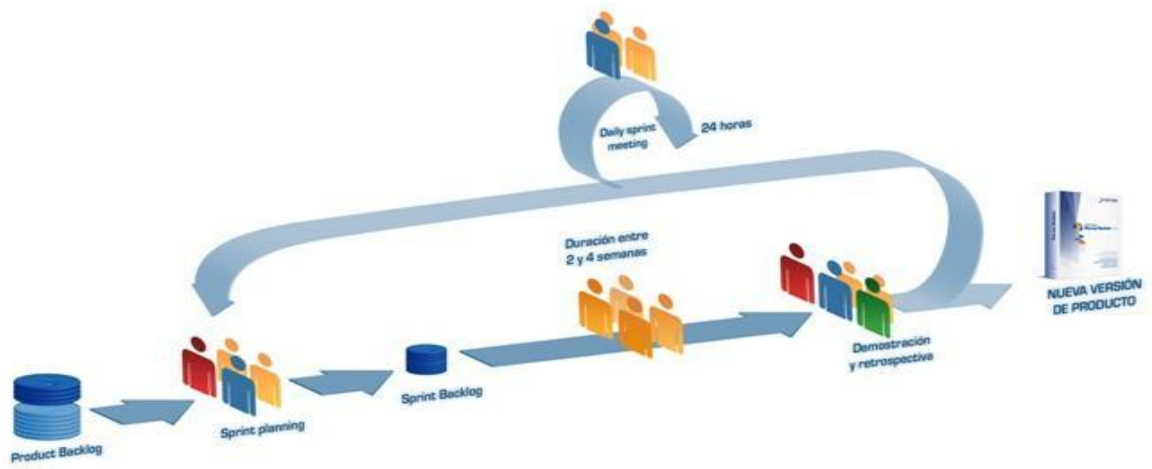


Fig 3-2 Proceso de Scrum
Fuente: Scrum Alliance

CAPÍTULO 4 : DESARROLLO DE LA PROPUESTA

Para el desarrollo de este proyecto se utilizó la metodología de desarrollo Scrum, esta es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para el cliente. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación.

Para el desarrollo de la aplicación se utilizó la práctica de programación TDD (*Test Driven Development*), esto es un desarrollo guiado por pruebas de software.

El TDD consiste en desarrollar pruebas, codificar y refactorizar el código construido.

A diferencia del procedimiento que usamos habitualmente, esto es construir el código y después realizar las pruebas unitarias, TDD establece que primero hay que realizar una prueba y a continuación desarrollar el código que la resuelve. Por lo tanto se desarrolla solo el código necesario y no se desperdician líneas de código.

A continuación se describen los pasos utilizados para el desarrollo de la propuesta aplicando la metodología SCRUM.

4.1. Generación del Product Backlog

Una vez finalizadas las entrevistas se ha llegado a la conclusión de que la Clínica San Bartolomé no cuenta con una Aplicación Distribuida que controle el inventario de medicinas, esta labor se la realiza de forma manual lo que provoca inconsistencias en la existencia de medicinas dentro del inventario.

Para la elaboración de la Aplicación Distribuida para el control del inventario se realizaron reuniones con el personal encargado del inventario de medicinas. En estas reuniones se establecieron los requerimientos con los que debe cumplir el sistema para solucionar los problemas que se han venido presentando lo largo de los años.

Con toda la información obtenida después de las entrevistas realizadas se procedió a realizar las historias de usuario de acuerdo a la metodología SCRUM.

Historias de usuario

Las historias de usuario son representaciones de requisitos de software escritas en una o dos frases utilizando el lenguaje común del usuario, es decir en un lenguaje nada técnico.

Historias de usuario del sistema

Historia de Usuario	
Numero: 1	Nombre: Ingreso de Laboratorio
Prioridad: Alta	Modificación de historia No.: No aplica
Descripción: Es necesario registrar a los laboratorios que proveen la medicina a la clínica, para el registro de un laboratorio se requiere el ruc del laboratorio, nombre del laboratorio, representante legal, dirección, teléfono que son datos importantes de un laboratorio.	

Tabla 4-1 Historia de Usuario de Ingreso de Laboratorio
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 2	Nombre: Ingreso Tipos de Usuario
------------------	----------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Para el administrador de la clínica es necesario registrar los tipos de usuarios, para el registro de un tipo de usuario es necesario registrar el nombre del tipo de usuario y una descripción.

Tabla 4-2 Historia de Usuario de Ingreso Tipos de Usuario
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 3	Nombre: Ingreso de Usuarios que utilizan la aplicación
------------------	--------------------------------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Es necesario registrar los usuarios que utilizarán la aplicación para lo cual es necesario ingresar los siguientes datos: cédula, nombre, apellido, dirección, teléfono y un e-mail, además debe estar relacionado a un tipo de usuario.

Tabla 4-3 Historia de Usuario de Ingreso de Usuarios que utilizan la aplicación.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 4	Nombre: Ingreso de Visitadores Médicos
------------------	----------------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Se requiere realizar mediante la aplicación el ingreso de visitadores médicos, para lo cual es necesario registrar los siguientes datos: cédula, nombre, apellido, dirección teléfono, e-mail.

Tabla 4-4 Historia de Usuario de Ingreso de Visitadores Médicos.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 5	Nombre: Ingreso de Tipo Presentación para la medicina
------------------	-------------------------------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Para el ingreso de medicinas es necesario registrar la presentación en la que viene la medicina, para lo cual se requiere ingresar los tipos de presentación, para esto es necesario ingresar los siguientes datos: nombre y una descripción.

Tabla 4-5 Historia de Usuario de Ingreso de Tipo Presentación para la medicina.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 6	Nombre: Ingreso de Destinos de la medicina
------------------	--------------------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: El usuario requiere registrar el lugar a donde se dirigirá la medicina para un mejor control del inventario, para lo cual es necesario registrar el nombre y una descripción.

Tabla 4-6 Historia de Usuario de Ingreso de Destinos de la medicina.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 7	Nombre: Ingreso de Medicinas
------------------	------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Es necesario para el usuario realizar el registro de medicinas para lo cual es necesario ingresar los siguientes datos de la medicina: el nombre medicinal, el nombre genérico, stock que en el primer ingreso será 0, el laboratorio al que pertenece la medicina, el laboratorio que provee la medicina y el tipo de presentación en la que viene la medicina.

Tabla 4-7 Historia de Usuario de Ingreso de Medicinas.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 8	Nombre: Registro de Egresos de Medicina
------------------	-----------------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Para controlar el inventario es necesario realizar egresos de la medicina por lo tanto se requiere registrar los siguientes datos en el egreso: la fecha del egreso y el usuario que realiza el egreso. Al momento de realizar egreso de medicinas se puede realizar el egreso de varias medicinas a la vez, para lo cual es necesario registrar los siguientes datos: la medicina a la cual se desea realizar el egreso, la cantidad de medicina, el lugar a donde va dirigida la medicina y una descripción.

Tabla 4-8 Historia de Usuario de Registro de Egresos de Medicina.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 9	Nombre: Registro de Ingresos de Medicina
------------------	------------------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Para controlar el inventario es necesario realizar ingresos de la medicina por lo tanto se requiere registrar los siguientes datos en el ingreso: la fecha de ingreso y el usuario que realiza el ingreso. Al momento de realizar ingreso de medicinas se puede ingresar varias medicinas en el mismo ingreso, para lo cual es necesario registrar los siguientes datos: la medicina a la cual se desea realizar el ingreso, la cantidad de medicina, una descripción y la fecha de caducidad.

Tabla 4-9 Historia de Usuario de Registro de Ingresos de Medicina.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 10	Nombre: Menú de la aplicación
-------------------	-------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Para un acceso fácil a las pantallas de la aplicación el usuario necesita de un menú en el que se encuentre todos los accesos a la aplicación.

Tabla 4-10 Historia de Usuario de Menú de la aplicación.
Elaborado por: Juan Santamaría

Historia de Usuario

Numero: 11	Nombre: Reportes de medicinas
-------------------	-------------------------------

Prioridad: Alta	Modificación de historia No.: No aplica
------------------------	-----------------------------------------

Descripción: Requiere que el sistema ofrezca opciones para generar reportes de medicinas que se encuentran en el inventario, los reportes que se necesitan son: Reporte General de Medicinas (Todo el inventario) Reporte por Medicina Reporte de Medicinas Caducadas.

Tabla 4-11 Historia de Usuario de Reporte de medicinas
Elaborado por: Juan Santamaría

El Product Backlog queda conformado con todas estas historias de usuario. El próximo paso en el proceso de SCRUM es el Sprint Planning.

4.2. Sprint Planning

Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint.

Para el desarrollo de esta aplicación se realizaron Sprint Planning cada cuatro semanas, cada Sprint varía de acuerdo a la complejidad en el desarrollo de cada historia de usuario.

4.3. Sprints

A continuación se describen todos los Sprints que ayudaron en el desarrollo del proyecto.

4.3.1. Primer Sprint

Dentro de este Sprint se realizaron las primeras tres historias de usuario:

- 1- Ingreso de Laboratorios
- 2- Ingreso de Tipos de Usuario
- 3- Ingreso de Usuarios

Cada historia de usuario tiene un Sprint Backlog que es el que contiene las tareas necesarias para que la historia de usuario quede terminada.

a. Sprint Backlog para Ingreso de Laboratorios

Tareas:

- Crear la tabla Laboratorios.
- Método para el ingreso de laboratorio desarrollado mediante TDD.
- Método para la modificación de un laboratorio ya existente desarrollado mediante TDD.
- Método para el guardado de un nuevo o modificado laboratorio desarrollado mediante TDD.
- Método para la eliminación de un laboratorio desarrollado mediante TDD.
- Desarrollo de la Interfaz para la manipulación de laboratorios.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

b. Implementación

En la implementación se procede a realizar el desarrollo de las tareas que contiene cada historia de usuario, a continuación se describe el desarrollo de las tareas.

Método para el ingreso de laboratorio:

Método creado en el controlador.

```
52
53 public void nuevo() {
54     laboratorioDataManager.setRuc(null);
55     laboratorioDataManager.setNombreComercial(null);
56     laboratorioDataManager.setRepresentanteLegal(null);
57     laboratorioDataManager.setDirección(null);
58     laboratorioDataManager.setTeleno(null);
59     laboratorioDataManager.setDescripción(null);
60     laboratorioDataManager.setLaboratorioSeleccionado(new Laboratorio());
61 }
62
```

Fig 4-1 Método para el ingreso de laboratorio
Elaborado por: Juan Santamaría

Método para la modificación de un laboratorio ya existente:

Método creado en el controlador.

```
105
106 public void editar() {
107     laboratorioDataManager.setRuc(laboratorioDataManager
108         .getLaboratorioSeleccionado().getRuc());
109     laboratorioDataManager.setNombreComercial(laboratorioDataManager
110         .getLaboratorioSeleccionado().getNombreComercial());
111     laboratorioDataManager.setRepresentanteLegal(laboratorioDataManager
112         .getLaboratorioSeleccionado().getRepresentanteLegal());
113     laboratorioDataManager.setDirección(laboratorioDataManager
114         .getLaboratorioSeleccionado().getDireccion());
115     laboratorioDataManager.setTeleno(laboratorioDataManager
116         .getLaboratorioSeleccionado().getTelefono());
117     laboratorioDataManager.setDescripción(laboratorioDataManager
118         .getLaboratorioSeleccionado().getDescripcion());
119 }
120
```

Fig 4-2 Método para la modificación de un laboratorio
Elaborado por: Juan Santamaría

Métodos para el guardado de un nuevo o modificado laboratorio:

Método creado en el controlador. (Ver Fig 4.3)

```
63 public void guardar() {
64     try {
65         laboratorioDataManager.getLaboratorioSeleccionado().setRuc(
66             laboratorioDataManager.getRuc());
67         laboratorioDataManager.getLaboratorioSeleccionado()
68             .setNombreComercial(
69                 laboratorioDataManager.getNombreComercial());
70         laboratorioDataManager.getLaboratorioSeleccionado()
71             .setRepresentanteLegal(
72                 laboratorioDataManager.getRepresentanteLegal());
73         laboratorioDataManager.getLaboratorioSeleccionado().setDireccion(
74             laboratorioDataManager.getDireccion());
75         laboratorioDataManager.getLaboratorioSeleccionado().setTelefono(
76             laboratorioDataManager.getTelefono());
77         laboratorioDataManager.getLaboratorioSeleccionado().setDescripcion(
78             laboratorioDataManager.getDescripción());
79
80         laboratorioBeanLocal.guardarLaboratorio(laboratorioDataManager
81             .getLaboratorioSeleccionado());
82         cargarDatos();
83         addInfoMessage("Registro guardado correctamente");
84     } catch (Exception e) {
85         addErrorMessage("Error al amacemar el Registro");
86     }
87 }
88
```

Fig 4-3 Método para guardar (controlador)
Elaborado por: Juan Santamaría

El método guardar creado en el controlador llama al método guardar laboratorio creado en el servicio:

```
29
30 @Override
31 public void guardarLaboratorio(Laboratorio laboratorio) {
32     laboratorioDao.guardarLaboratorio(laboratorio);
33 }
34
```

Fig 4-4 Método para guardar (servicio)
Elaborado por: Juan Santamaría

El método guardar laboratorio creado en el servicio llama al método guardar laboratorio creado en el servicio de acceso a la base de datos:

```
27
28 @Override
29 public void guardarLaboratorio(Laboratorio laboratorio) {
30     if (laboratorio.getId() == null) {
31         em.persist(laboratorio);
32     } else {
33         em.merge(laboratorio);
34     }
35 }
```

Fig 4-5 Método para guardar (servicio de acceso a la base de datos)
Elaborado por: Juan Santamaría

Prueba de unidad para verificar el correcto funcionamiento del método guardar:
(Ver Fig 4-6)

```

20
21 // Test para guardar
22 @Test
23 public void testGuardarLaboratorio() {
24
25     Long codigo = null;
26     Laboratorio nuevoLaboratorio = new Laboratorio();
27     nuevoLaboratorio.setRuc("1804166534001");
28     nuevoLaboratorio.setNombreComercial("Nombre ejemplo");
29     nuevoLaboratorio.setRepresentanteLegal("Representante ejemplo");
30     nuevoLaboratorio.setDireccion("Direccion ejemplo");
31
32     laboratorioBeanRemote.guardarLaboratorio(nuevoLaboratorio);
33
34     codigo = nuevoLaboratorio.getId();
35
36     nuevoLaboratorio = null;
37
38     nuevoLaboratorio = laboratorioBeanRemote.obtenerPorId(codigo);
39
40     Assert.assertNotNull(nuevoLaboratorio);
41
42 }

```

Fig 4-6 Prueba de unidad método guardar
Elaborado por: Juan Santamaría

Métodos para la eliminación de un laboratorio:

Método creado en el controlador.

```

94
95 public void eliminar() {
96     try {
97         laboratorioBeanLocal.eliminarLaboratorio(laboratorioDataManager
98             .getLaboratorioEliminar().getId());
99         cargarDatos();
100        addInfoMessage("Registro eliminado correctamente");
101    } catch (Exception e) {
102        addErrorMessage("Error al eliminar el Registro");
103    }
104 }

```

Fig 4-7 Método para eliminar (controlador)
Elaborado por: Juan Santamaría

El método eliminar creado en el controlador llama al método eliminar laboratorio creado en el servicio:

```

34
35 @Override
36 public void eliminarLaboratorio(Long idLaboratorio) {
37     laboratorioDao.eliminarLaboratorio(idLaboratorio);
38 }
39

```

Fig 4-8 Método para eliminar (servicio)
Elaborado por: Juan Santamaría

El método eliminar laboratorio creado en el servicio llama al método eliminar laboratorio creado en el servicio de acceso a la base de datos: (Ver Fig 4.9)

```
36
37 @Override
38 public void eliminarLaboratorio(Long idLaboratorio) {
39     em.remove(em.find(Laboratorio.class, idLaboratorio));
40 }
..
```

Fig 4-9 Método para eliminar (servicio de acceso a la BDD)
Elaborado por: Juan Santamaría

Desarrollo de la Interfaz para la manipulación:

En esta tarea de la historia de usuario se desarrolla la interfaz para la manipulación de datos:

The screenshot shows a web application window titled "LABORATORIO". At the top left, there is a message log with options for "Error message", "Fatal message", "Info message", and "Warning message". Below this is a section titled "INGRESO DE DATOS" (Data Entry) containing several input fields with labels: "Ruc", "Nombre Comercial", "Representante Legal", "Dirección", "Teléfono", and "Descripción". Each field has a corresponding placeholder text starting with "#{" followed by a variable name like "laboratorioDM.ruc". Below the input fields are three buttons: "Nuevo", "Guardar", and "Cancelar". At the bottom of the window is a table with the following structure:

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
#{laboratorio.ruc}	#{laboratorio.nombreComercial}	#{laboratorio.representanteLegal}	#{laboratorio.direccion}	#{laboratorio.telefono}	#{laboratorio.descripcion}	

Fig 4-10 Interfaz de manipulación de datos
Elaborado por: Juan Santamaría

Después de que se haya desarrollado la interfaz de manipulación de datos, la siguiente tarea es la validación de campos requeridos que se describe a continuación:

```
30         <h:inputText value="#{laboratorioDM.dirección}" required="true"
31             requiredMessage="Campo Direccion requerido" />
32         <h:outputLabel value="Teléfono" />
33         <h:inputText value="#{laboratorioDM.teleno}" />
34         <h:outputLabel value="Descripción" />
35         <h:inputText value="#{laboratorioDM.descripcion}" />
36     </h:panelGrid>
37 <br />
38 <a4j:commandButton value="Nuevo"
39     action="#{laboratorioController.nuevo()}" render="panelIngreso"
40     style="font-size: 14px;" execute="@this"></a4j:commandButton>
41 <h:outputLabel value=" " />
42 <a4j:commandButton value="Guardar"
43     action="#{laboratorioController.guardar()}"
44     render="panelIngreso,tableDatos" style="font-size: 14px;"></a4j:commandButton>
45 <h:outputLabel value=" " />
46 <a4j:commandButton value="Cancelar"
47     action="#{laboratorioController.init()}" render="panelIngreso"
48     style="font-size: 14px;" execute="@this"></a4j:commandButton>
49 <p />
50 <a4j:outputPanel id="tableDatos">
51     <rich:dataTable id="tableLaboratorio"
52         value="#{laboratorioDM.laboratorios}" var="laboratorio"
53         rendered="#{laboratorioDM.laboratorios ne null and not empty laboratorioDM.laboratorios}"
54         rows="6">
```

Fig 4-11 Validación de campos requeridos

Elaborado por: Juan Santamaría

Una vez desarrolladas todas las tareas de la historia de usuario es necesario generar el WAR de la aplicación y desplegarla utilizando dicho WAR, estos procesos se describen a continuación.

c. Generación del WAR de la aplicación

Esta aplicación es una aplicación Maven por lo tanto al compilar la aplicación se genera un archivo de tipo WAR el cual contiene las clases y páginas que conforman la aplicación, además de las librerías necesarias para su correcto funcionamiento, para la generación del WAR se siguieron los siguientes pasos:

- 1- En una ventana de comando dirigirse al directorio en donde se encuentra el código fuente del cual se va a generar el WAR.
- 2- Ejecutar el siguiente comando: `mvn clean install`

Con esto el empaquetado WAR se ha generado en la carpeta TARGET del directorio donde se encuentra el código fuente de la aplicación.

d. Desplegar el WAR de la aplicación

Para desplegar (deploy) el WAR de la aplicación generado, se debe copiar el archivo WAR en la carpeta /standalone/deployments del servidor de aplicaciones.

Debido a que el proceso utilizado en la metodología SCRUM es cíclico y repetitivo se describió el proceso completo solamente para la primera historia de usuario, pero el proceso se lo debe repetir hasta terminar con el desarrollo de todas las historias de usuario.

A continuación se describen los Sprints restantes con sus historias de usuario y las tareas correspondientes.

Sprint Backlog de Ingreso de Tipos de Usuario

Tareas:

- Crear la tabla Tipos de Usuario.
- Método para el ingreso de un tipo de usuario desarrollado mediante TDD.
- Método para la modificación de un tipo de usuario ya existente desarrollado mediante TDD.
- Método para el guardado de un nuevo o modificado tipo de usuario desarrollado mediante TDD.
- Método para la eliminación de un tipo de usuario desarrollado mediante TDD.
- Desarrollo de la Interfaz para la manipulación de tipos de usuario.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Sprint Backlog para Ingreso de Usuarios

Tareas:

- Crear la tabla Usuarios con relación de hijo a la tabla Tipos de Usuario
- Método para el ingreso de un usuario desarrollado mediante TDD.
- Método para la modificación de un usuario ya existente desarrollado mediante TDD.
- Método para el guardado de un nuevo o modificado usuario desarrollado mediante TDD.
- Método para la eliminación de un usuario desarrollado mediante TDD.
- Desarrollo de la Interfaz para la manipulación de usuarios.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

El siguiente paso dentro de la metodología SCRUM el siguiente paso es el *demo* en el cual se realiza una demostración al usuario de todo lo desarrollado en el Sprint.

Demo

Al finalizar este Sprint se realizó una reunión con el usuario para mostrarle las historias de usuario alcanzadas en el Sprint, lo que se le mostró en la reunión fue lo siguiente:

Interfaz de laboratorios:

The screenshot shows the 'INVENTARIO MEDICINAS' web application. At the top left is a logo with the letters 'SB' inside a blue cross. To the right of the logo is the text 'INVENTARIO MEDICINAS'. Below this is a navigation bar with three items: 'Administración', 'Control Inventario', and 'Reportes'. The main content area is titled 'LABORATORIO' and contains a form for 'INGRESO DE DATOS'. The form has several input fields for 'Ruc', 'Nombre Comercial', 'Representante Legal', 'Dirección', 'Teléfono', and 'Descripción'. Below the form are three buttons: 'Nuevo', 'Guardar', and 'Cancelar'. At the bottom of the form is a table with the following data:

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
444	LabComa	LanRep	Quito	666	NN	 

Fig 4-12 Interfaz de Laboratorios
Elaborado por: Juan Santamaría

Interfaz de Tipo de Usuario

INVENTARIO MEDICINAS

Administración Control Inventario Reportes

TIPO USUARIO

INGRESO DE DATOS

Nombre

Descripción

Nuevo Guardar Cancelar

Nombre	Descripción	Acciones
Administrador	Usuario Administrador	

Fig 4-13 Interfaz de Tipos de Usuario
Elaborado por: Juan Santamaría

Interfaz de Usuario

INVENTARIO MEDICINAS

Administración Control Inventario Reportes

USUARIO

INGRESO DE DATOS

Cédula

Nombre

Apellido

Dirección

Teléfono

E-mail

Tipo Usuario Seleccione...

Nuevo Guardar Cancelar

Cédula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
144	JJSL	SS	QQ	222	NN	Administrador	

Fig 4-14 Interfaz de Usuarios
Elaborado por: Juan Santamaría

4.3.2. Segundo Sprint

Dentro de este sprint se realizaron las siguientes historias de usuario:

- 4- Ingreso de Visitadores Médicos
- 5- Ingreso de Tipo Presentación para la medicina
- 6- Ingreso de Destinos de la medicina

Sprint Backlog para Ingreso de Visitadores Médicos

Tareas:

- Crear la tabla Visitadores Médicos.
- Método para el ingreso de un visitador médico desarrollado mediante TDD.

- Método para la modificación de un visitador médico ya existente desarrollado mediante TDD.
- Método para el guardado de un nuevo o modificado visitador médico desarrollado mediante TDD.
- Método para la eliminación de un visitador médico mediante TDD.
- Desarrollo de la Interfaz para la manipulación de visitadores médicos.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Sprint Backlog para Ingreso de Tipo de Presentación para la medicina

Tareas:

- Creación de la tabla Tipos de Presentación.
- Método para el ingreso de un tipo de presentación desarrollado mediante TDD.
- Método para la modificación de un tipo de presentación ya existente desarrollado mediante TDD.
- Método para el guardado de un nuevo o modificado registro de un tipo de presentación mediante TDD.
- Método para la eliminación de un tipo de presentación mediante TDD.
- Desarrollo de la Interfaz para la manipulación de tipos de presentación para la medicina.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Sprint Backlog para Ingreso de Destinos para la medicina

Tareas:

- Crear la tabla Destinos para la medicina.
- Método para el ingreso de un destino de medicina desarrollado mediante TDD.
- Método para la modificación de un destino de medicina ya existente desarrollado mediante TDD.
- Método para el guardado de un nuevo o modificado registro de destino de medicina mediante TDD.
- Método para la eliminación de un destino de medicina mediante TDD.
- Desarrollo de la Interfaz para la manipulación de destinos de medicina.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Demo

Al finalizar este Sprint se realizó una reunión con el usuario para mostrarle las historias de usuario alcanzadas en el Sprint, lo que se le mostró en la reunión fue lo siguiente:

Interfaz de Visitador

Cédula	Nombre	Apellido	Dirección	Teléfono	Email	Descripción	Acciones
123	JJ	SS	QQ	222	NN	desc	
123	JJSzL	SS	QQ	222	NN	desc	

Fig 4-15 Interfaz de Visitadores
Elaborado por: Juan Santamaría

Interfaz de Tipo Presentación de la medicina

Nombre	Descripcion	Acciones
Gotas	Son gotas	

Fig 4-16 Interfaz de Tipos Presentación de Medicinas
Elaborado por: Juan Santamaría

Interfaz de Destino de Medicina

Administración Control Inventario Reportes

DESTINO MEDICINA

INGRESO DE DATOS

Nombre

Descripción

Nuevo Guardar Cancelar

Nombre	Descripción	Acciones
Consultorio	Consultorio del medico	

Fig 4-17 Interfaz de Destino de Medicina
Elaborado por: Juan Santamaría

4.3.3. Tercer Sprint

Dentro de este sprint se realizaron las siguientes historias de usuario:

- 7- Ingreso de medicina
- 8- Registro de un egreso de medicina

Sprint Backlog para Ingreso de Medicina

Tareas:

- Crear la tabla Medicinas.
- Método para la carga de ítems de laboratorios que serán cargados en el combo de laboratorios, desarrollado mediante TDD.
- Método para la carga de ítems de tipos de presentación para medicinas que serán cargados en el combo de tipos de presentación, desarrollado mediante TDD.
- Método para la carga de ítems de visitadores que serán cargados en el combo de visitadores, desarrollado mediante TDD.
- Método para la modificación de una medicina ya existente, desarrollado mediante TDD.
- Método para guardar la medicina creada, desarrollado mediante TDD.
- Método para la eliminación de una medicina, desarrollado mediante TDD.

- Desarrollo de la interfaz para la manipulación de medicinas.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Sprint Backlog para registro de un egreso de medicina

Tareas:

- Crear la tabla para el registro de los Egresos de medicina.
- Método para la carga de ítems de usuarios que serán cargados en el combo de usuarios desarrollado mediante TDD.
- Método que carga la fecha actual en el componente de selección de fecha.
- Método agregar medicinas en el detalle del egreso, desarrollado mediante TDD.
- Método para la carga de ítems de medicinas que serán cargados en el combo de medicinas, desarrollado mediante TDD.
- Método para la carga de ítems de destinos de medicina que serán cargados en el combo de destinos de medicinas, desarrollado mediante TDD.
- Método para guardar el egreso de medicina creado, guardando primero el maestro y a continuación el detalle, desarrollado mediante TDD.
- Desarrollo de la interfaz para la creación de un egreso de medicina.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Demo

Al finalizar este Sprint se realizó una reunión con el usuario para mostrarle las historias de usuario alcanzadas en el Sprint, lo que se le mostró en la reunión fue lo siguiente:

Interfaz de Medicina



The screenshot shows the 'INVENTARIO MEDICINAS' interface. It features a navigation bar with 'Administración', 'Control Inventario', and 'Reportes'. The main content area is titled 'MEDICINA' and contains a form for 'INGRESO DE DATOS'. The form fields are: 'Nombre Medicinal', 'Nombre Comercial', 'Tipo Presentacion' (dropdown), 'Visitador' (dropdown), 'Laboratorio' (dropdown), and 'Stock'. Below the form are buttons for 'Nuevo', 'Guardar', and 'Cancelar'. A table below the form lists existing medicines:

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Lapa	Laparamicina	Gotas	JJ	LabComa	10	 

Fig 4-18 Interfaz de Medicinas
Elaborado por: Juan Santamaría

Interfaz de Egreso de Medicina



The screenshot shows the 'EGRESO MEDICINAS' interface. It features a navigation bar with 'Administración', 'Control Inventario', and 'Reportes'. The main content area is titled 'EGRESO MEDICINAS' and contains a 'Nuevo Egreso' button. Below the button is a table of disbursements:

Usuario	Fecha Egreso	Acciones
JJSL	2013-12-05 00:00:00.0	 
JJSL	2013-12-05 00:00:00.0	 

Fig 4-19 Interfaz de Egreso de Medicina 1
Elaborado por: Juan Santamaría



Administración Control Inventario Reportes

EGRESO MEDICINAS

INGRESO DE DATOS

Usuario

Fecha

Medicina	Cantidad	Destino	Descripcion
<input type="text" value="Selecione..."/>	0	<input type="text" value="Selecione..."/>	<input type="text"/>

Fig 4-20 Interfaz de Egreso de Medicina 2
Elaborado por: Juan Santamaría

4.3.4. Cuarto Sprint

Dentro de este Sprint se realizaron las siguientes historias de usuario:

- 9- Registro de Ingreso de Medicina
- 10- Menú de la aplicación
- 11- Reporte de medicinas

Sprint Backlog para registro de un ingreso de medicina

Tareas:

- Crear la tabla para el registro de los Ingresos de medicina.
- Método para la carga de ítems de usuario que serán cargados en el combo de usuarios desarrollado mediante TDD.
- Método que carga la fecha actual en el componente de selección de fecha.
- Método para la carga de ítems de medicinas que serán cargados en el combo de medicinas, desarrollado mediante TDD.
- Método para guardar el ingreso de medicina creado, guardando primero el maestro y a continuación el detalle, desarrollado mediante TDD.
- Desarrollo de la interfaz para la creación de un ingreso de medicina.
- Desarrollo de validaciones de campos requeridos en pantalla.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Sprint Backlog para el menú de la aplicación

Tareas:

- Crear una tabla para la administración del menú.
- Método para la generación dinámica del menú, desarrollado mediante TDD.
- Desarrollo de la interfaz para el menú.

Sprint Backlog para el menú de la aplicación

Tareas:

- Método para la carga de ítems de medicinas que serán cargadas al combo de medicinas para el reporte por medicinas desarrollado mediante TDD.
- Método para la generación de reporte general de inventario de medicinas desarrollado mediante TDD.
- Método para la generación de reporte por medicina desarrollado mediante TDD.
- Método para la generación de reporte de medicinas caducadas desarrollado mediante TDD.
- Desarrollo de la interfaz para la generación de reporte general de inventario de medicinas.
- Desarrollo de la interfaz para la generación de reporte por medicina.
- Desarrollo de la interfaz para la generación de reporte de medicinas caducadas.

Todas las tareas fueron desarrolladas en el orden establecido anteriormente y de acuerdo a los requerimientos de la historia de usuario.

Demo

Al finalizar este Sprint se realizó una reunión con el usuario para mostrarle las historias de usuario alcanzadas en el Sprint, lo que se le mostró en la reunión fue lo siguiente:

Interfaz de Ingreso de Medicina



The screenshot shows the 'INGRESO MEDICINAS' interface. At the top left is the 'SB' logo. To the right, the text 'INVENTARIO MEDICINAS' is displayed. Below this is a navigation bar with 'Administración', 'Control Inventario', and 'Reportes'. The main content area is titled 'INGRESO MEDICINAS' and contains a 'Nuevo Ingreso' button. Below the button is a table with the following data:

Usuario	Fecha Ingreso	Acciones
JJSL	2013-12-04 00:00:00.0	 
JJSL	2013-12-05 00:00:00.0	 

Fig 4-21 Interfaz de Ingreso de Medicina 1
Elaborado por: Juan Santamaría



The screenshot shows the 'INGRESO MEDICINAS' interface for adding new data. It features the 'SB' logo and 'INVENTARIO MEDICINAS' text. The navigation bar includes 'Administración', 'Control Inventario', and 'Reportes'. The main area is titled 'INGRESO MEDICINAS' and contains a 'Nuevo Ingreso' button. Below this is the 'INGRESO DE DATOS' section with 'Usuario' (a dropdown menu) and 'Fecha' (a date picker). An 'Agregar Medicina' button is located below the form. Underneath is a table for adding medicine details:

Medicina	Cantidad	Fecha Caducidad	Descripcion
Selecione...	0	ene 9, 2014	

At the bottom of the table are 'Guardar' and 'Cancelar' buttons.

Fig 4-22 Interfaz Ingreso de Medicina 2
Elaborado por: Juan Santamaría

Menú de la Aplicación

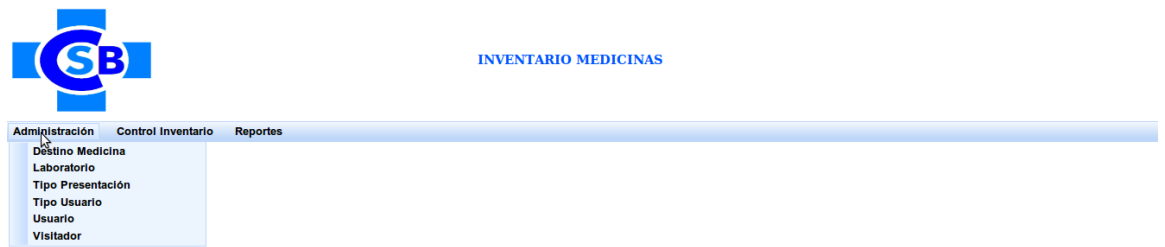


Fig 4-23 Menú de la aplicación 1
Elaborado por: Juan Santamaría

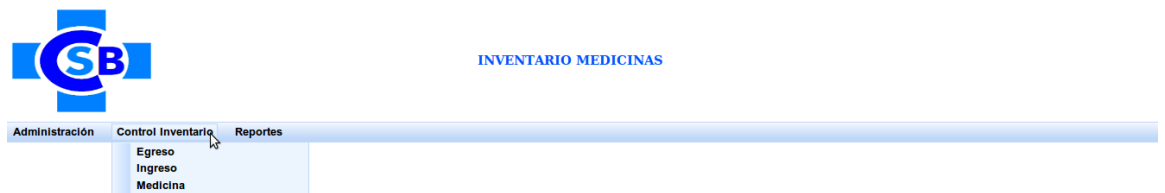


Fig 4-24 Menú de la aplicación 2
Elaborado por: Juan Santamaría

Sección de Reportes

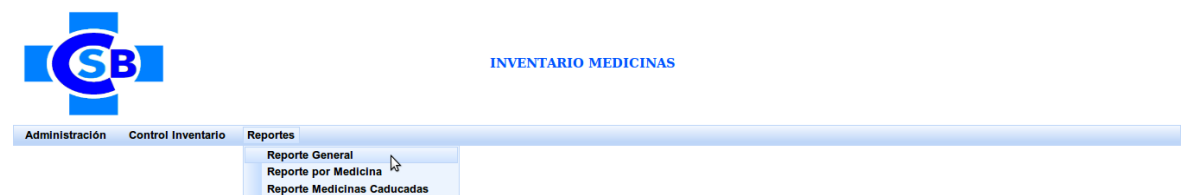


Fig 4-25 Sección de Reportes
Elaborado por: Juan Santamaría



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

REPORTE INVENTARIO GENERAL

INGRESO DE DATOS
Fecha de Corte: abr 16, 2014

Generar Exportar

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock
Laparona	Laparamicina	Gotas	JJ	LabComa	9
Tirotricina	Bucoctricina	Gotas	Juan	LabComa	3
Peroxido de Hidrogeno	Agua Oxigenada	Gotas	JJ	LabComa	4
Hidroxido de Magnesio	Magnecia	Gotas	Juan	LabComa	5

Fig 4-26 Reporte Inventario General
Elaborado por: Juan Santamaría



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

REPORTE INVENTARIO POR MEDICINA

INGRESO DE DATOS
Fecha de Corte: abr 18, 2014
Medicina: Bucoctricina

Generar Exportar

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock
Tirotricina	Bucoctricina	Gotas	Juan	LabComa	3

Fig 4-27 Reporte por Medicina
Elaborado por: Juan Santamaría



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

REPORTE MEDICINAS CADUCADAS

Generar Exportar

Lote	Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Cantidad	Fecha Caducidad	Costo \$
1	Laparona	Laparamicina	Gotas	10	2013-12-04 00:00:00.0	65
51	Laparona	Laparamicina	Gotas	2	2013-12-05 00:00:00.0	72
52	Laparona	Laparamicina	Gotas	8	2013-12-05 00:00:00.0	16
151	Laparona	Laparamicina	Gotas	3	2014-01-07 00:00:00.0	75

Fig 4-28 Reporte Medicinas Caducadas
Elaborado por: Juan Santamaría

4.4. Versión final de la Aplicación

Una vez desarrolladas todas las historias de usuario se generó una versión final de la aplicación de acuerdo a lo que se propone en SCRUM, esta contiene todas funcionalidades ya mostradas al usuario de la aplicación en los demos realizados al final de cada Sprint.

4.5. Modelo de la base de datos

Debido a que se aplicó la metodología SCRUM la base de datos se fue generando progresivamente de acuerdo a los requerimientos de cada historia de usuario, esta base de datos fue creada sobre el sistema de gestión de base de datos postgres.

El modelo de la base de datos resultante que se ajusta a los requerimientos que la aplicación necesita es el siguiente: (*Ver figura 4.25*)

4.6. Implantación de la aplicación

Antes de poner a funcionar la aplicación en la clínica es necesario realizar configuraciones previas.

4.6.1. Requerimientos de Software

Para que la aplicación de Control de Inventario de Medicinas pueda ser ejecutada correctamente, se debe tener instalados los siguientes requerimientos de software:

- Ambiente de desarrollo de aplicaciones Java: Java SE Development Kit 6u30 (Java JDK version 1.6)
- Servidor de aplicaciones para desplegar la aplicación: JBoss Application Server 7.1.1.Final
- Seleccionar el link para descargar el zip o tar.gz de la version 7.1.1 Final (Everything (NOT Java EE6 Certified))
- Motor de base de datos: PostgreSQL 9.1.2-1
- Herramienta de administración de proyectos: Maven 3.0.4

4.6.2. Configuración del servidor de aplicaciones

Se debe configurar en el servidor JBOSS lo siguiente:

Configuración del Data Source

Se debe agregar el origen de datos a donde se conectara nuestra aplicación dentro del tag <datasources></datasources> en el archivo /standalone/configuration/standalone.xml del servidor de aplicaciones.

Editar la dirección IP, puerto, nombre de usuario y contraseña de acuerdo a la instalación del motor de base de datos PostgreSQL correspondiente.

Además agregar el driver correspondiente a PostgreSQL dentro del tag <drivers></drivers>.

(Ver Fig 4.27)

4.6.3. Cronograma de Implantación

Actividades	Tiempo en días						
	1	2	3	4	5	6	7
Instalación de requerimientos de software en la maquina que va a hacer de servidor de BDD.	X						
Instalación de requerimientos de software en la maquina que va a hacer de servidor de aplicaciones.	X						
Importación de la BDD.		X					
Configuraciones del servidor de aplicaciones.		X					
Levantamiento de la aplicación.			X				
Capacitación a usuarios de la aplicación.				X	X	X	X
Entrega formal de la aplicación al administrador.							X

Tabla 4-12 Cronograma de Implantación
Elaborado por: Juan Santamaría

4.6.4. Cronograma de Capacitación

Páginas	Tiempo en días			
	1	2	3	4
PÁGINA DE ADMINISTRACIÓN DE DESTINO DE MEDICINA	X			
PAGINA ADMINISTRACION DE LABORATORIOS	X			
PAGINA ADMINISTRACION DE TIPO PRESENTACION	X			
PAGINA ADMINISTRACION DE TIPO USUARIO		X		
PAGINA ADMINISTRACION DE USUARIO		X		
PAGINA ADMINISTRACION DE VISITADOR		X		
PAGINA ADMINISTRACION DE MEDICINA			X	
PAGINA ADMINISTRACION DE EGRESOS DE MEDICINA			X	
PAGINA ADMINISTRACION DE INGRESOS DE MEDICINA			X	
PAGINA REPORTE INVENTARIO GENERAL				X
PAGINA REPORTE INVENTARIO POR MEDICINA				X
PAGINA REPORTE INVENTARIO MEDICINA CADUCADA				X

Tabla 4-13 Cronograma de Capacitación
Elaborado por: Juan Santamaría

Modelo Relacional de Base de Datos

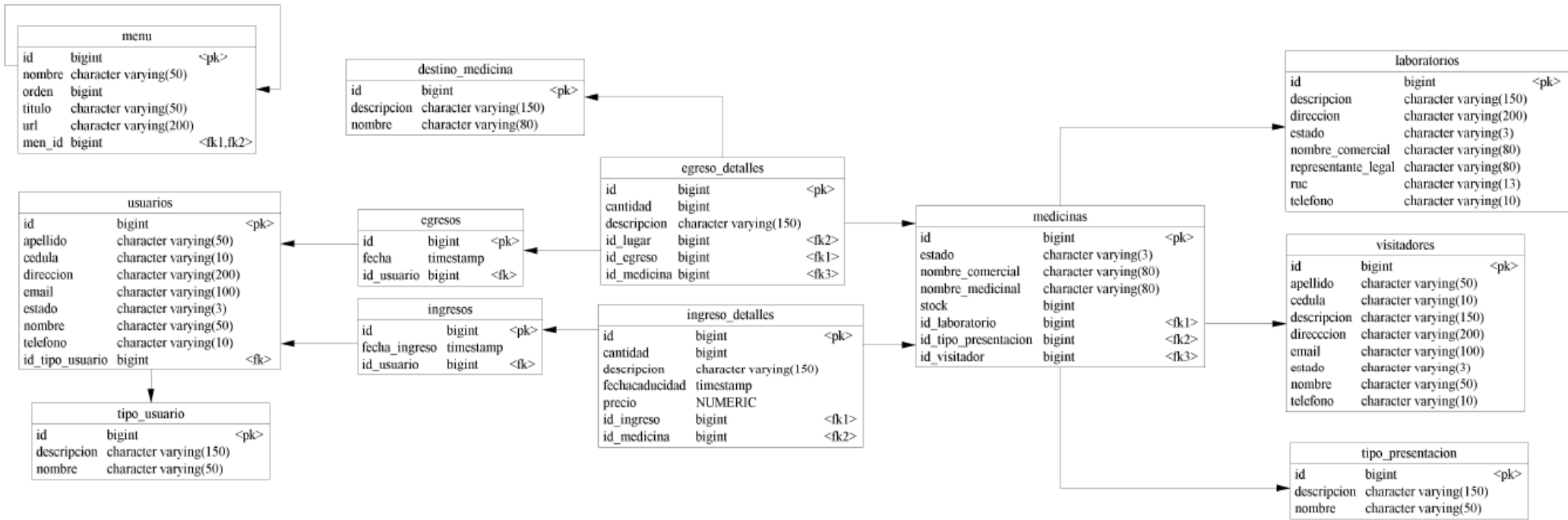


Fig 4-29 Modelo Relacional Base de Datos
Elaborado por: Juan Santamaría

Modelo Entidad Relación de Base de Datos

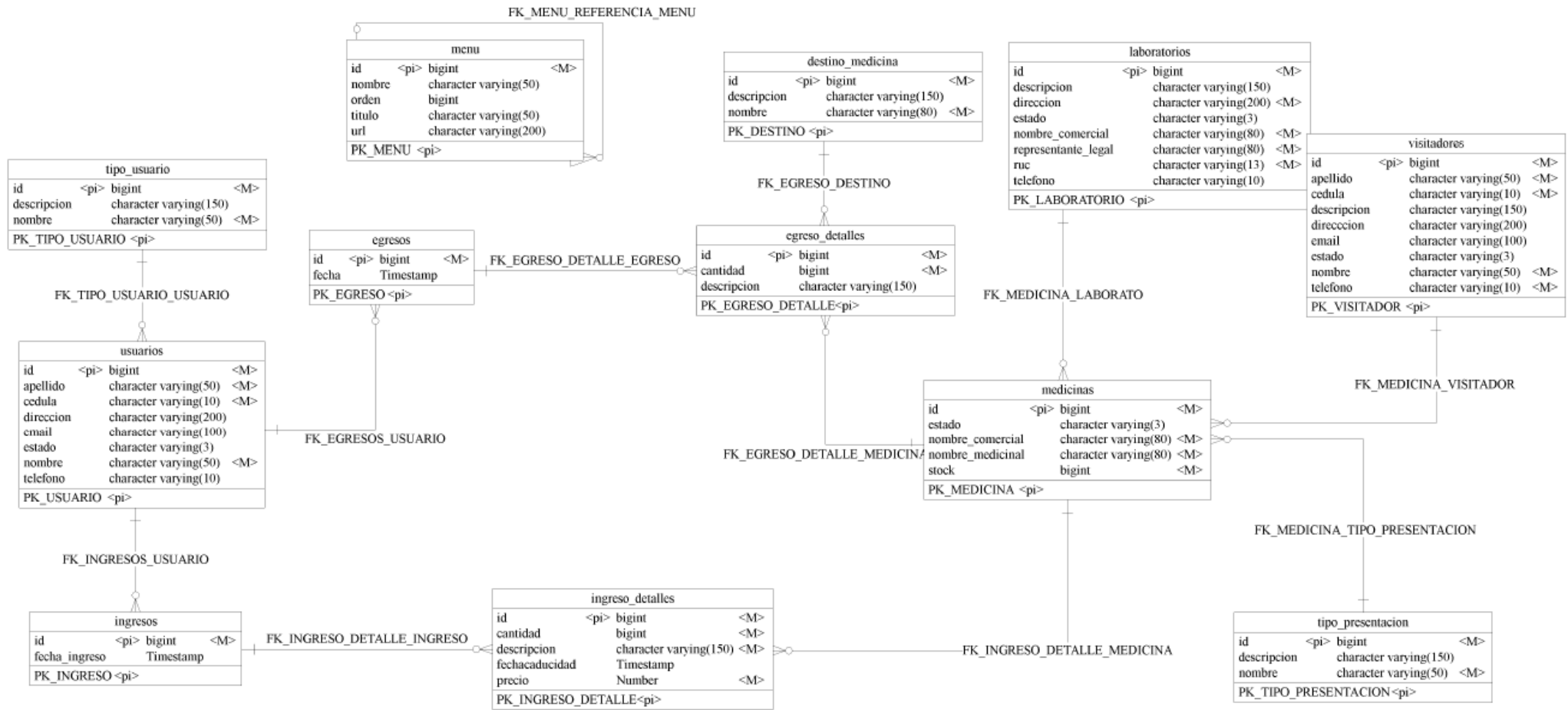


Fig 4-30 Modelo Entidad Relacion de Base de Datos
Elaborado por: Juan Santamaría

Diccionario de datos

Lista de Atributos de la tabla destino_medicina

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Columna que representa la clave primaria de la tabla
descripcion	character varying(150)		Campo que almacena la descripción del registro
nombre	character varying(80)	X	Campo que almacena el nombre del registro

Tabla 4-14 Lista de Atributos de la tabla destino_medicina

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla egreso_detalle

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
cantidad	bigint	X	Campo que almacena el número de medicinas que se va a egresar
descripcion	character varying(150)		Campo que almacena una descripción del registro

Tabla 4-15 Lista de Atributos de la tabla egreso_detalle

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla egresos

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
fecha	Timestamp		Campo que almacena la fecha del egreso

Tabla 4-16 Lista de Atributos de la tabla egresos

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla ingreso_detalle

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
cantidad	bigint	X	Campo que almacena el numero de medicinas que se desea ingresar
descripcion	character varying(150)	X	Campo que almacena la descripcion del ingreso
fechacaducidad	Timestamp		Campo que almacena la fecha de caducidad de la medicina a ingresar
precio	Numeric	X	Campo que almacena el precio de la medicina

Tabla 4-17 Lista de Atributos de la tabla ingreso_detalle

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla ingresos

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
fecha_ingreso	Timestamp		Campo que almacena la fecha de ingreso

Tabla 4-18 Lista de Atributos de la tabla ingresos

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla laboratorios

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
descripcion	character varying(150)		Campo que almacena la descripcion del registro
direccion	character varying(200)	X	Campo que almacena la direccion de laboratorio

estado	character varying(3)		Campo que almacena el estado ACT o INA del registro
nombre_comercial	character varying(80)	X	Campo que almacena el nombre comercial del laboratorio
representante_legal	character varying(80)	X	Campo que almacena al representante legal del laboratorio
ruc	character varying(13)	X	Campo que almacena el ruc del laboratorio
telefono	character varying(10)		Campo que almacena el teléfono del laboratorio

Tabla 4-19 Lista de Atributos de la tabla laboratorios

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla medicinas

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
estado	character varying(3)		Campo que almacena el estado ACT o INA del registro
nombre_comercial	character varying(80)	X	Campo que almacena el nombre comercial de la medicina
nombre_medicinal	character varying(80)	X	Campo que almacena el nombre medicinal de la medicina
stock	bigint	X	Campo que almacena a la cantidad de medicinas que hay en stock de ese registro

Tabla 4-20 Lista de Atributos de la tabla medicinas

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla menú

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
nombre	character varying(50)		Campo que almacena al nombre de la opcion en el menu
orden	bigint		Campo que almacena la posicion de la opcion en el menu
titulo	character varying(50)		Campo que almacena al titulo de la opcion en el menu
url	character varying(200)		Campo que almacena la url hacia donde redirecciona la opcion en el menu

Tabla 4-21 Lista de Atributos de la tabla menú

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla tipo_presentacion

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
descripcion	character varying(150)		Campo que almacena una descripcion del registro
nombre	character varying(50)	X	Campo que almacena el nombre del registro

Tabla 4-22 Lista de Atributos de la tabla tipo_presentacion

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla tipo_usuario

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
descripcion	character varying(150)		Campo que almacena la descripcion del registro
nombre	character varying(50)	X	Campo que almacena el nombre del registro

Tabla 4-23 Lista de Atributos de la tabla tipo_usuario

Elaborado por: Juan Santamaría

Lista de Atributos de la tabla usuarios

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
apellido	character varying(50)	X	Campo que almacena al apellido del usuario
cedula	character varying(10)	X	Campo que almacena a la cedula del usuario
direccion	character varying(200)		Campo que almacena la direccion del usuario
email	character varying(100)		Campo que almacena el e-mail del usuario
estado	character varying(3)		Campo que almacena el estado ACT o INA del registro
nombre	character varying(50)	X	Campo que almacena el nombre del usuario
telefono	character varying(10)		Campo que almacena el telefono del usuario

Tabla 4-24 Lista de Atributos de la tabla usuarios
Elaborado por: Juan Santamaría

Lista de Atributos de la tabla visitantes

Nombre	Tipo de Dato	Not Null	Comentario
id	bigint	X	Campo que almacena la clave primaria de la tabla
apellido	character varying(50)	X	Campo que almacena el apellido del visitador
cedula	character varying(10)	X	Campo que almacena la cedula del visitador
descripcion	character varying(150)		Campo que almacena una descripcion del registro
direccion	character varying(200)		Campo que almacena la direccion del visitador
email	character varying(100)		Campo que almacena el e-mail del visitador
estado	character varying(3)		Campo que almacena el estado ACT o INA del registro

nombre	character varying(50)	X	Campo que almacena el nombre del visitador
telefono	character varying(10)	X	Campo que almacena el teléfono del visitador

Tabla 4-25 Lista de Atributos de la tabla visitantes

Elaborado por: Juan Santamaría

Configuración Datasource

```

standalone.xml
<subsystem xmlns="urn:jboss:domain:datasources:1.0">
  <datasources>
    <datasource jndi-name="java:jboss/datasources/ExampleDS" pool-name="ExampleDS" enabled="true" use-java-context="true">
      <connection-url>jdbc:h2:mem:test;DB_CLOSE_DELAY=-1</connection-url>
      <driver>h2</driver>
      <security>
        <user-name>sa</user-name>
        <password>sa</password>
      </security>
    </datasource>
    <datasource jta="true" jndi-name="java:jboss/datasources/inventarioDS" pool-name="inventarioDS_pool" enabled="true" use-java-context="true" use-
ccm="true">
      <connection-url>jdbc:postgresql://localhost:5432/inventario</connection-url>
      <driver>postgresql</driver>
      <pool>
        <min-pool-size>10</min-pool-size>
        <max-pool-size>100</max-pool-size>
        <prefill>true</prefill>
        <use-strict-min>false</use-strict-min>
        <flush-strategy>FailingConnectionOnly</flush-strategy>
      </pool>
      <security>
        <user-name>postgres</user-name>
        <password>postgres</password>
      </security>
    </datasource>
  </datasources>
  <drivers>
    <driver name="h2" module="com.h2database.h2">
      <xa-datasource-class>org.h2.jdbcx.JdbcDataSource</xa-datasource-class>
    </driver>
    <driver name="postgresql" module="org.postgresql">
      <xa-datasource-class>org.postgresql.xa.PGXADataSource</xa-datasource-class>
    </driver>
  </drivers>
</datasources>
XML Ancho de la tabulación: 8 Ln 95, Col 11 INS

```

Fig 4-31 Configuración Datasource

Elaborado por: Juan Santamaría

4.7. Arquitectura de la aplicación distribuida

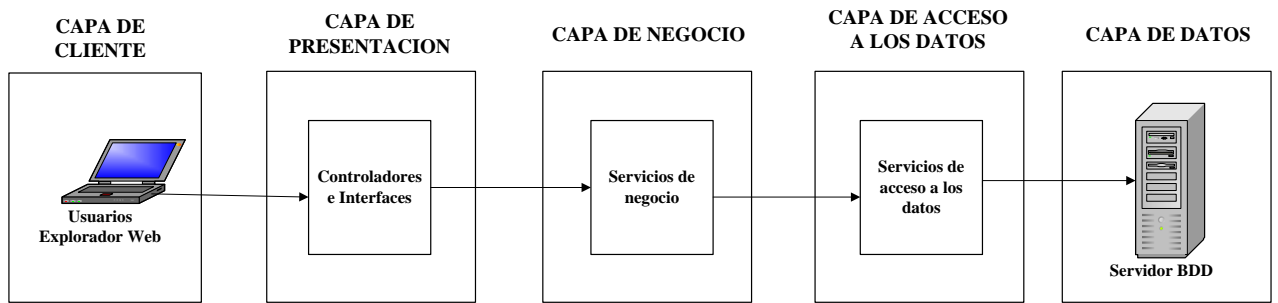


Fig 4-32 Arquitectura de la aplicación distribuida
Elaborado por: Juan Santamaría

CAPÍTULO 5 : CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

Luego de la elaboración de la Aplicación Distribuida basada en las especificaciones JEE6 para el control del inventario de medicinas en la Clínica San Bartolomé se concluye que:

- Para mejorar el control de inventario es necesario mejorar el nivel de calidad sobre los procesos que intervienen en este.
- La administradora del inventario de medicinas de Clínica San Bartolomé ha indicado que con el sistema se han disminuido las inconsistencias y que los reportes generados por la aplicación ayudan a reducir notablemente el tiempo de la toma física.
- Se concluye que la Clínica San Bartolomé necesita contar con la tecnología necesaria para el control de inventario de medicinas.
- Para el proyecto se utilizaron herramientas de software libre, tales como Postgres, JBoss, Maven, que facilitaron la elaboración del mismo, estas no representan gastos en su utilización, además que existe una amplia información sobre su funcionamiento y aplicación.

- Desarrollar la aplicación bajo las especificaciones JEE6 fue de gran ayuda, puesto que permite la estructuración del proyecto de mejor manera y por lo tanto la aplicación está mejor organizada lo que facilitara en el futuro su mantenimiento.
- La selección de una metodología de desarrollo ágil como es SCRUM fue una parte muy importante, ya que con esta metodología se realizan demostraciones del software al cliente continuamente y así el usuario aporta con ideas, correcciones, conocimiento de procesos durante el desarrollo de la aplicación, lo cual garantiza la satisfacción del cliente en el producto final.
- Para el desarrollo de la aplicación se utilizó la práctica de programación TDD, lo cual fue muy útil ya que con esta práctica se garantiza un código de programación más limpio puesto que contiene solamente el código necesario, esto será servirá al momento de realizar un mantenimiento, debido a que el código se encuentra bien refactorizado y es de fácil entendimiento.
- La utilización del gestor de BDD PostgreSQL permite en el manejo de una gran cantidad de datos, además el nexo que tiene con el servidor de aplicaciones JBoss encaja dentro de las especificaciones JEE6, complementándose de buena manera en el desarrollo del proyecto.
- Una vez implantada la aplicación se pudo confirmar que el servidor de aplicaciones JBoss es muy estable en su funcionamiento, es de fácil configuración y no requiere de una máquina con grades recursos.

5.2. Recomendaciones

- La aplicación fue diseñada para que se pueda añadir nuevos módulos sin mayor complejidad, por lo tanto se recomienda que en un futuro se añada nuevos módulos dentro de la aplicación, tales como facturación y control de historias clínicas con la cual se completara la automatización de todos los procesos.
- Se recomienda seguir las especificaciones JEE6 para la elaboración de una aplicación distribuida ya que con estas se puede obtener una aplicación mucho más organizada y robusta, y con la selección de las herramientas adecuadas facilitan el desarrollo de las aplicaciones.

- Al utilizar Maven para la compilación del proyecto se evita incluir las dependencias en la estructura misma del proyecto debido a que esta herramienta aporta un sistema de gestión de dependencias basado en repositorios, por lo tanto es muy recomendable su utilización en proyectos relacionados con JEE6.
- Se recomienda utilizar la metodología de desarrollo ágil SCRUM para el desarrollo de futuros proyectos dentro de la FISEI debido a que con esta metodología existe una relación más directa con el cliente lo cual facilita el desarrollo de los proyectos.

Bibliografía

- [1] Jorge Olavarrieta de la Torre, «Control de Inventario,» 7 Junio 1999. [En línea]. Disponible en:
http://books.google.com.ec/books?id=EXzhFaRE9rUC&pg=PA43&lpg=PA43&dq=control+de+inventario&source=bl&ots=L_ju_0yNfd&sig=SU6bQhemOpnApCtqE2xTjF1TnM&hl=es&sa=X&ei=TXFqUPmQI6ux0QGHxYGQAAQ&ved=0CDEQ6AEwAQ
- [2] Mikell P. Groover, «Control de Inventario,» 25 Enero 2002. [En línea]. Disponible en:
<http://books.google.com.ec/books?id=tcV0l37tUr0C&pg=PA988&lpg=PA988&dq=control+de+inventario&source=bl&ots=7XOD93bvRu&sig=2HTVdQWDaZD2Rd0E4sEa06OfJ6M&hl=es&sa=X&ei=TXFqUPmQI6ux0QGHxYGQAAQ&ved=0CD0Q6AEwAw#v=onepage&q=control%20de%20inventario&f=false>
- [3] Santi Caballé, Fatos Xhafa, «Aplicación Distribuida,» 30 Mayo 2010. [En línea]. Disponible en:
http://books.google.com.ec/books?id=8bZ5gd9zPYkC&printsec=frontcover&dq=aplicaciones+distribuidas&source=bl&ots=8qvCjSfI8a&sig=mo_U5ssRaDZMhj5HzorE3DWoxq0&hl=es&sa=X&ei=V7pOUNisEono0QGoq4DIDg&sqi=2&ved=0CC4Q6AEwAA#v=onepage&q=aplicaciones%20distribuidas&f=false
- [4] Francisco Javier Moliner López, «Aplicación Distribuida,» 29 Marzo 2011. [En línea]. Disponible en:
<http://books.google.com.ec/books?id=WKr3qQ6RW6EC&pg=PA263&lpg=PA263&dq=aplicaciones+distribuidas&source=bl&ots=AIOsbAwgCS&sig=FJF--d5zVe1abcyyUk7dSHSkoKc&hl=es&sa=X&ei=V7pOUNisEono0QGoq4DIDg&sqi=2&ved=0CFoQ6AEwCA#v=onepage&q=aplicaciones%20distribuidas&f=false>
- [5],[6] Jorge Sales, «Características de una Aplicación Distribuida, Ventajas de una Aplicación Distribuida,». [En línea]. Disponible en:
http://www3.uji.es/~mmarques/e16/trabajos/t_jorge_sales.pdf
- [7] Eduardo Suarez Galvez, «Arquitectura de aplicación distribuida,» Febrero 23 de 2013. [En línea]. Disponible en:
http://arquitecturaaplicacionesdistribuidas.blogspot.com/2013/02/unidad-2-arquitectura-de-aplicaciones_22.html
- [8] PostgreSQL Global Development Group, «PostgreSQL,» 1996-2014. [En línea]. Disponible en:
<http://www.postgresql.org/about/>
- [9] J. Martín Sierra, Programador Certificado Java 2, México: Alfaomega Grupo Editor, 2010.

- [10] Ed Ort, «Especificaciones JEE6,» Diciembre 29. [En línea]. Available: <http://www.oracle.com/technetwork/articles/javace/javaee6overview-141808.html>
- [11] JBoss Community, «Servidor de Aplicaciones JBoss,». [En línea]. Disponible en: <http://www.jboss.org/jbossas>
- [12] Apache Maven Project, «Apache Maven,» 2002-2014. [En línea]. Disponible en: <http://maven.apache.org/>
- [13] JasperSoft Community, «JasperReports,» 2000-2014. [En línea]. Disponible en: <http://community.jaspersoft.com/project/jasperreports-library>
- [14] Masadelante.com, «JasperReports,». [En línea], Disponible en: <http://www.masadelante.com/faqs/css>
- [15] Softeng Software Engineers, «Procesos y Roles de Scrum,». [En línea]. Disponible en: <http://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum/proceso-roles-de-scrum.html>

Anexos

Anexo 1: Manual de Usuario

MANUAL DE USUARIO

**“APLICACIÓN DISTRIBUIDA BASADO EN LAS
ESPECIFICACIONES JEE6 PARA EL CONTROL DEL
INVENTARIO DE MEDICINAS EN LA CLÍNICA SAN
BARTOLOMÉ”**

**OBJETIVO: EL OBJETIVO DE ESTE MANUAL ES DE
PROPORCIONAR UNA GUÍA A LOS USUARIOS PARA
FACILITAR EL USO DE LA APLICACIÓN DISTRIBUIDA PARA
EL CONTROL DEL INVENTARIO DE MEDICINAS**

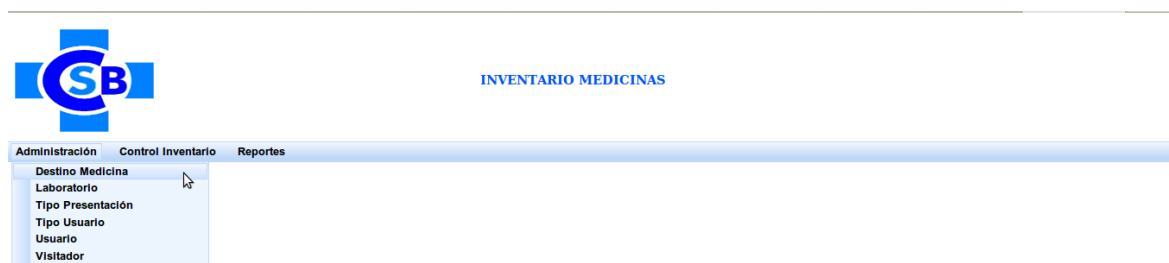
INGRESO A LA APLICACIÓN

Para ingresar a la aplicación se lo debe hacer mediante un Browser, en el browser ingresar la dirección IP de la maquina en donde se encuentra la aplicación, accediendo por el puerto 8080, ejemplo: 192.168.1.5:8080/InventarioClinicaWeb una vez ingresada esta dirección la aplicación aparecerá en el browser.



PÁGINA DE ADMINISTRACIÓN DE DESTINO DE MEDICINA

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Administración > Destino Medicina.



Al ingresar en esta opción se mostrará la siguiente página:

En esta página constan todos los destinos de medicinas existentes, además de los botones de administración.

Ingreso de un nuevo destino de medicina

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de destino de medicina, mostrando un mensaje de éxito.



Administración Control Inventario Reportes

DESTINO MEDICINA

Registro guardado correctamente

INGRESO DE DATOS

Nombre Quirofano

Descripción Quirofano

Nuevo Guardar Cancelar

Nombre	Descripción	Acciones
Consultorio	Consultorio del medico	[Icono] [Icono]
Quirofano	Quirofano	[Icono] [Icono]

Eliminación de un registro existente de destino de medicina

- Click en el botón Eliminar del registro que se desea eliminar.
- La aplicación mostrará un mensaje de confirmación.



Administración Control Inventario Reportes

DESTINO MEDICINA

INGRESO DE DATOS

Nombre

Descripción

Nuevo Guardar

Nombre	Descripción	Acciones
Consultorio	Consultorio del medico	[Icono] [Icono]
Quirofano	Quirofano	[Icono] [Icono]

La página en http://localhost:8080 dice:

Esta seguro de que desea eliminar el registro ?

Cancelar Aceptar

- Si confirma la eliminación el registro se eliminará, mostrando un mensaje de éxito.



Administración Control Inventario Reportes

DESTINO MEDICINA

Registro eliminado correctamente

INGRESO DE DATOS

Nombre

Descripción

Nuevo Guardar Cancelar

Nombre	Descripción	Acciones
Consultorio	Consultorio del medico	[Icono] [Icono]

Edición de un registro existente

- Click en el botón Editar del registro que se desea editar.

- La aplicación cargará la información del registro seleccionado.



The screenshot shows the 'INVENTARIO MEDICINAS' application interface. At the top left is a logo with the letters 'SB' inside a blue cross. The title 'INVENTARIO MEDICINAS' is centered at the top. Below the title is a navigation bar with three tabs: 'Administración', 'Control Inventario', and 'Reportes'. The main content area is titled 'DESTINO MEDICINA' and contains a form for data entry. The form has two input fields: 'Nombre' with the value 'Consultorio' and 'Descripción' with the value 'Consultorio del medico'. Below the input fields are three buttons: 'Nuevo', 'Guardar', and 'Cancelar'. At the bottom of the form is a table with three columns: 'Nombre', 'Descripción', and 'Acciones'. The table contains one row with the values 'Consultorio', 'Consultorio del medico', and a set of action icons (edit, delete).

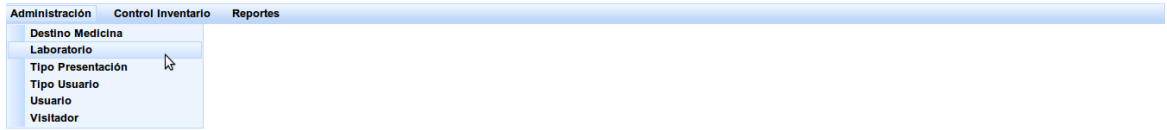
- Editar los campos que se deseen.
- Click en el botón guardar para que se actualice el registro seleccionado.
- La aplicación mostrará un mensaje de éxito de guardado.



This screenshot is identical to the previous one, but it includes a success message at the top of the form area: 'Registro guardado correctamente'. The rest of the interface, including the navigation bar, the 'DESTINO MEDICINA' title, the input fields, buttons, and the table, remains the same.

PAGINA ADMINISTRACION DE LABORATORIOS

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Administración > Laboratorio.



Al ingresar en esta opción se mostrará la siguiente página:



En esta página constan todos los laboratorios existentes, además de los botones de administración.

Ingreso de un nuevo laboratorio

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

LABORATORIO

Campo Ruc requerido
 Campo Nombre Comercial requerido
 Campo Representante Legal requerido
 Campo Dirección requerido

INGRESO DE DATOS

Ruc

Nombre Comercial

Representante Legal

Dirección

Teléfono

Descripción

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
444	LabComa	LanRep	Quito	666	NN	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de laboratorio, mostrando un mensaje de éxito.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

LABORATORIO

Registro guardado correctamente

INGRESO DE DATOS

Ruc

Nombre Comercial

Representante Legal

Dirección

Teléfono

Descripción

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
444	LabComa	LanRep	Quito	666	NN	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
1804166534	Laboratorio X	Representante legal	Ambato	123456789	Sin descripción	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Eliminación de un registro existente de laboratorio

- Click en el botón Eliminar del registro que se desea eliminar.
- La aplicación mostrará un mensaje de confirmación.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

LABORATORIO

INGRESO DE DATOS

Ruc

Nombre Comercial

Representante Legal

Dirección

Teléfono

Descripción

Nuevo Guardar Cancelar

Esta seguro de que desea eliminar el registro ?

Cancelar Aceptar

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
444	LabComa	LanRep	Quito	666	NN	
1804166534	Laboratorio X	Representante legal	Ambato	123456789	Sin descripción	

- Si confirma la eliminación el registro se eliminará, mostrando un mensaje de éxito.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

LABORATORIO

Registro eliminado correctamente

INGRESO DE DATOS

Ruc

Nombre Comercial

Representante Legal

Dirección

Teléfono

Descripción

Nuevo Guardar Cancelar

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
444	LabComa	LanRep	Quito	666	NN	

Edición de un registro existente

- Click en el botón Editar del registro que se desea editar.
- La aplicación cargará la información del registro seleccionado.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

LABORATORIO

INGRESO DE DATOS

Ruc 444

Nombre Comercial LabComa

Representante Legal LanRep

Dirección Quito

Teléfono 222

Descripción nn

Nuevo Guardar Cancelar

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
444	LabComa	LanRep	Quito	123456789	Sin descripción	

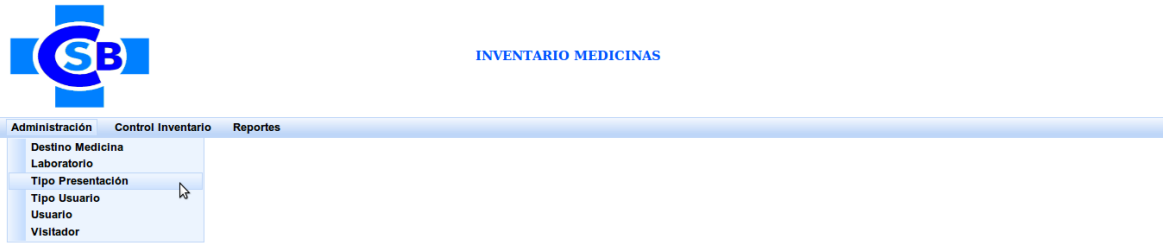
- Editar los campos que se deseen.
- Click en el botón guardar para que se actualice el registro seleccionado.
- La aplicación mostrará un mensaje de éxito de guardado.

The screenshot shows the 'INVENTARIO MEDICINAS' application interface. At the top left is the 'SB' logo. The main title is 'INVENTARIO MEDICINAS'. Below the title is a navigation bar with 'Administración', 'Control Inventario', and 'Reportes'. The central area displays a 'LABORATORIO' form with a success message: 'Registro guardado correctamente'. Below the message is the 'INGRESO DE DATOS' section with input fields for Ruc (444), Nombre Comercial (LabComa), Representante Legal (LanRep), Dirección (Quito), Teléfono (123456789), and Descripción (Sin descripción). At the bottom of the form are buttons for 'Nuevo', 'Guardar', and 'Cancelar'. Below the form is a table with the following data:

Ruc	Nombre Comercial	Representante Legal	Dirección	Teléfono	Descripción	Acciones
444	LabComa	LanRep	Quito	123456789	Sin descripción	

PAGINA ADMINISTRACION DE TIPO PRESENTACION

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Administración > Tipo Presentación.



Al ingresar en esta opción se mostrará la siguiente página:



TIPO PRESENTACION

INGRESO DE DATOS

Nombre

Descripcion

Nombre	Descripcion	Acciones
Gotas	Son gotas	<input type="button" value=""/> <input type="button" value="X"/>

En esta página constan todos los tipos de presentación existentes, además de los botones de administración.

Ingreso de un nuevo tipo presentación

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:



TIPO PRESENTACION

❗ Campo Nombre requerido

INGRESO DE DATOS

Nombre

Descripcion

Nombre	Descripcion	Acciones
Gotas	Son gotas	<input type="button" value=""/> <input type="button" value="X"/>

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de tipo presentación, mostrando un mensaje de éxito.



TIPO PRESENTACION

Registro guardado correctamente

INGRESO DE DATOS

Nombre

Descripcion

Nombre	Descripcion	Acciones
Gotas	Son gotas	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
Ampollas	Ampollas	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Eliminación de un registro existente de tipo presentación

- Click en el botón Eliminar del registro que se desea eliminar.
- La aplicación mostrará un mensaje de confirmación.



TIPO PRESENTACION

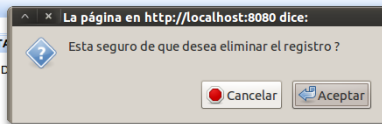
Registro guardado correctamente

INGRESO DE DATOS

Nombre

Descripcion

Nombre	Descripcion	Acciones
Gotas	Son gotas	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
Ampollas	Ampollas	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>



- Si confirma la eliminación el registro se eliminará, mostrando un mensaje de éxito.



TIPO PRESENTACION

Registro eliminado correctamente

INGRESO DE DATOS

Nombre

Descripcion

Nombre	Descripcion	Acciones
Gotas	Son gotas	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Edición de un registro existente

- Click en el botón Editar del registro que se desea editar.
- La aplicación cargará la información del registro seleccionado.



TIPO PRESENTACION

INGRESO DE DATOS

Nombre

Descripcion

Nombre	Descripcion	Acciones
Gotas	Son gotas	

- Editar los campos que se deseen.
- Click en el botón guardar para que se actualice el registro seleccionado.
- La aplicación mostrará un mensaje de éxito de guardado.



TIPO PRESENTACION

Registro guardado correctamente

INGRESO DE DATOS

Nombre

Descripcion

Nombre	Descripcion	Acciones
Gotas	Gotas	

PAGINA ADMINISTRACION DE TIPO USUARIO

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Administración > Tipo Usuario.



- Destino Medicina
- Laboratorio
- Tipo Presentación
- Tipo Usuario**
- Usuario
- Visitador

Al ingresar en esta opción se mostrará la siguiente página:



TIPO USUARIO

INGRESO DE DATOS

Nombre

Descripción

Nombre	Descripción	Acciones
Administrador	Administradores	

En esta página constan todos los tipos de usuario existentes, además de los botones de administración.

Ingreso de un nuevo tipo usuario

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:



TIPO USUARIO

❌ Campo Nombre requerido

INGRESO DE DATOS

Nombre

Descripción

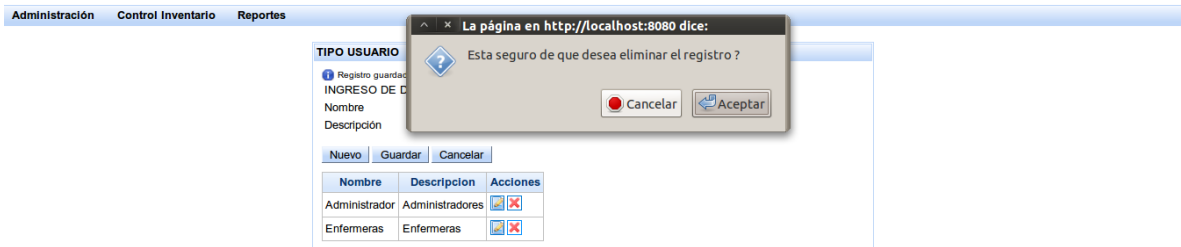
Nombre	Descripción	Acciones
Administrador	Administradores	

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de tipo usuario, mostrando un mensaje de éxito.



Eliminación de un registro existente de tipo usuario

- Click en el botón Eliminar del registro que se desea eliminar.
- La aplicación mostrará un mensaje de confirmación.



- Si confirma la eliminación el registro se eliminará, mostrando un mensaje de éxito.



Edición de un registro existente

- Click en el botón Editar del registro que se desea editar.

- La aplicación cargará la información del registro seleccionado.

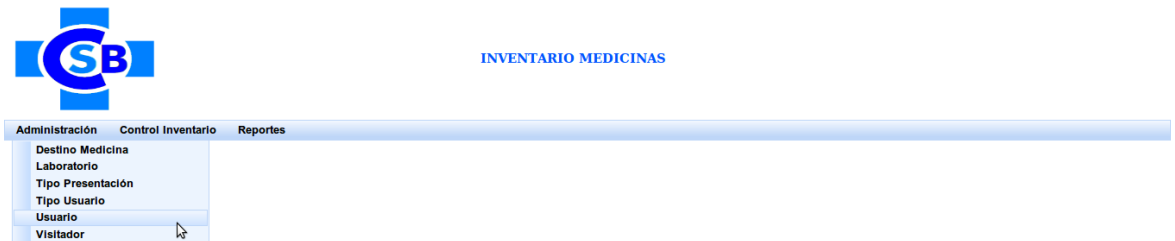


- Editar los campos que se deseen.
- Click en el botón guardar para que se actualice el registro seleccionado.
- La aplicación mostrará un mensaje de éxito de guardado.



PAGINA ADMINISTRACION DE USUARIO

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Administración > Usuario.



Al ingresar en esta opción se mostrará la siguiente página:

Administración Control Inventario Reportes

USUARIO

INGRESO DE DATOS

Cédula

Nombre

Apellido

Dirección

Teléfono

E-mail

Tipo Usuario Seleccione... ▾

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
144	JJSJ	SS	QQ	222	NN	Administrador	

En esta página constan todos los usuarios existentes, además de los botones de administración.

Ingreso de un nuevo usuario

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:

Administración Control Inventario Reportes

USUARIO

Campo Cedula requerido
 Campo Nombre requerido
 Campo Apellido requerido
 Campo Tipo Usuario requerido

INGRESO DE DATOS

Cédula

Nombre

Apellido

Dirección

Teléfono

E-mail

Tipo Usuario Seleccione... ▾

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
144	JJSJ	SS	QQ	222	NN	Administrador	

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.

- Al dar click en dicho botón se creara un nuevo registro de usuario, mostrando un mensaje de éxito.

USUARIO

Registro guardado correctamente

INGRESO DE DATOS

Cédula: 1804166534
 Nombre: Juan
 Apellido: Santamaria
 Dirección: Ambato
 Teléfono: 123456789
 E-mail: juan@mail.com
 Tipo Usuario: Administrador

Nuevo Guardar Cancelar

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
144	JJSL	SS	QQ	222	NN	Administrador	[Iconos]
1804166534	Juan	Santamaria	Ambato	123456789	juan@mail.com	Administrador	[Iconos]

Eliminación de un registro existente de usuario

- Click en el botón Eliminar del registro que se desea eliminar.
- La aplicación mostrará un mensaje de confirmación.

USUARIO

INGRESO DE D

Cédula
Nombre
Apellido
Dirección
Teléfono
E-mail
Tipo Usuario

Seleccione...

Nuevo Guardar Cancelar

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
144	JJSL	SS	QQ	222	NN	Administrador	[Iconos]
1804166534	Juan	Santamaria	Ambato	123456789	juan@mail.com	Administrador	[Iconos]

- Si confirma la eliminación el registro se eliminará, mostrando un mensaje de éxito.



USUARIO

Registro eliminado correctamente

INGRESO DE DATOS

Cédula

Nombre

Apellido

Dirección

Teléfono

E-mail

Tipo Usuario

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
144	JJSL	SS	QQ	222	NN	Administrador	

Edición de un registro existente

- Click en el botón Editar del registro que se desea editar.
- La aplicación cargará la información del registro seleccionado.



USUARIO

INGRESO DE DATOS

Cédula

Nombre

Apellido

Dirección

Teléfono

E-mail

Tipo Usuario

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
144	JJSL	SS	QQ	222	NN	Administrador	

- Editar los campos que se deseen.
- Click en el botón guardar para que se actualice el registro seleccionado.
- La aplicación mostrará un mensaje de éxito de guardado.



USUARIO

Registro guardado correctamente

INGRESO DE DATOS

Cédula: 1804166534
Nombre: JJSJL
Apellido: SS
Dirección: QQ
Teléfono: 222
E-mail: juan@mail.com
Tipo Usuario: Administrador

Nuevo Guardar Cancelar

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Tipo Usuario	Acciones
1804166534	JJSJL	SS	QQ	222	juan@mail.com	Administrador	

PAGINA ADMINISTRACION DE VISITADOR

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Administración > Visitador.



- Destino Medicina
- Laboratorio
- Tipo Presentación
- Tipo Usuario
- Usuario
- Visitador

Al ingresar en esta opción se mostrará la siguiente página:



VISITADOR

INGRESO DE DATOS

Cédula
Nombre
Apellido
Dirección
Teléfono
E-mail
Descripción

Nuevo Guardar Cancelar

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Descripción	Acciones
123	JJ	SS	QQ	222	NN	desc	
123	JJSzL	SS	QQ	222	nn	DESC	

En esta página constan todos los visitantes existentes, además de los botones de administración.

Ingreso de un nuevo visitante

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:

The screenshot shows the 'INVENTARIO MEDICINAS' application interface. At the top left is the 'SB' logo. The navigation bar includes 'Administración', 'Control Inventario', and 'Reportes'. The main content area is titled 'VISITADOR' and displays a form with the following fields: Cédula, Nombre, Apellido, Dirección, Teléfono, E-mail, and Descripción. Above the form, there are four red error messages: 'Campo Cedula requerido', 'Campo Nombre requerido', 'Campo Apellido requerido', and 'Campo Direccion requerido'. Below the form are buttons for 'Nuevo', 'Guardar', and 'Cancelar'. A table below the form shows existing records:

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Descripción	Acciones
123	JJ	SS	QQ	222	NN	desc	[icon] [icon]
123	JJSzL	SS	QQ	222	nn	DESC	[icon] [icon]

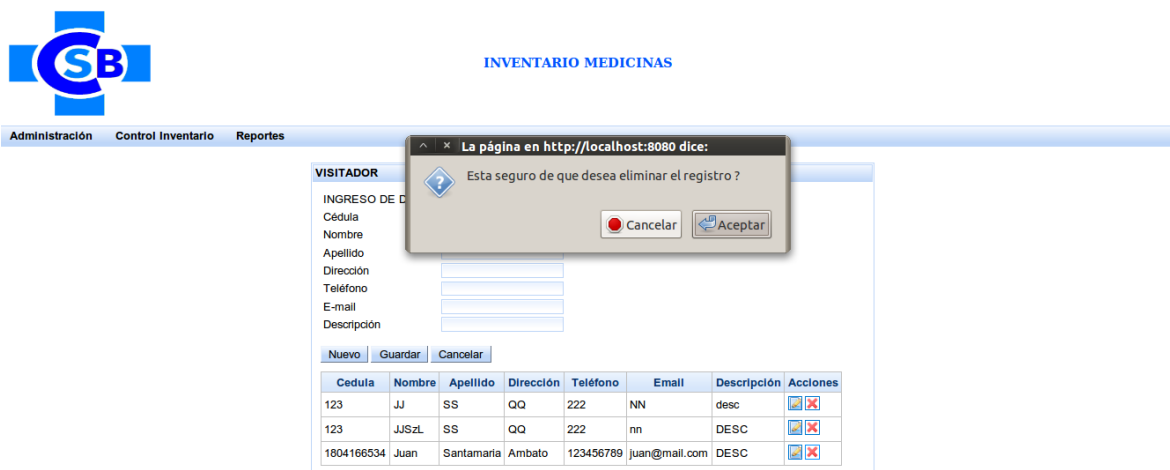
- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de visitante, mostrando un mensaje de éxito.

The screenshot shows the 'INVENTARIO MEDICINAS' application interface after a successful registration. The 'VISITADOR' section displays a blue message: 'Registro guardado correctamente'. Below this, the 'INGRESO DE DATOS' form shows the following filled-in fields: Cédula (1804166534), Nombre (Juan), Apellido (Santamaria), Dirección (Ambato), Teléfono (123456789), E-mail (juan@mail.com), and Descripción (DESC). Below the form are buttons for 'Nuevo', 'Guardar', and 'Cancelar'. The table below the form now includes the newly added record:

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Descripción	Acciones
123	JJ	SS	QQ	222	NN	desc	[icon] [icon]
123	JJSzL	SS	QQ	222	nn	DESC	[icon] [icon]
1804166534	Juan	Santamaria	Ambato	123456789	juan@mail.com	DESC	[icon] [icon]

Eliminación de un registro existente de visitador

- Click en el botón Eliminar del registro que se desea eliminar.
- La aplicación mostrará un mensaje de confirmación.



- Si confirma la eliminación el registro se eliminará, mostrando un mensaje de éxito.



Edición de un registro existente

- Click en el botón Editar del registro que se desea editar.
- La aplicación cargará la información del registro seleccionado.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

VISITADOR

INGRESO DE DATOS

Cédula 1804166534
Nombre Juan
Apellido SS
Dirección QQ
Teléfono 222
E-mail juan@mail.com
Descripción DESC

Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Descripción	Acciones
123	JJ	SS	QQ	222	NN	desc	
123	JJSZL	SS	QQ	222	nn	DESC	

- Editar los campos que se deseen.
- Click en el botón guardar para que se actualice el registro seleccionado.
- La aplicación mostrará un mensaje de éxito de guardado.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

VISITADOR

Registro guardado correctamente

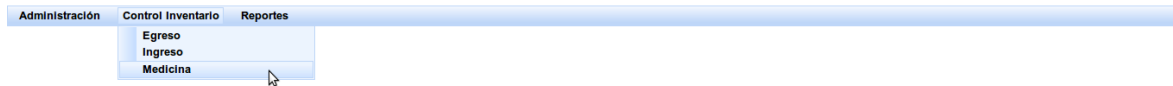
INGRESO DE DATOS

Cédula 1804166534
Nombre Juan
Apellido SS
Dirección QQ
Teléfono 222
E-mail juan@mail.com
Descripción DESC

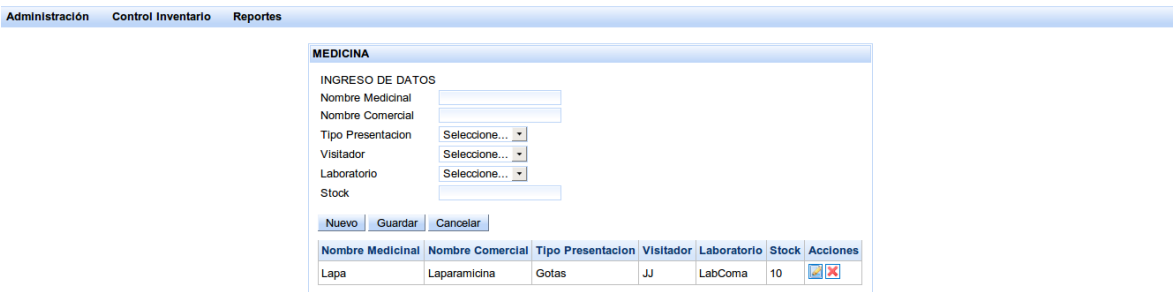
Cedula	Nombre	Apellido	Dirección	Teléfono	Email	Descripción	Acciones
123	JJ	SS	QQ	222	NN	desc	
1804166534	Juan	SS	QQ	222	juan@mail.com	DESC	

PAGINA ADMINISTRACION DE MEDICINA

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Control Inventario > Medicina.



Al ingresar en esta opción se mostrará la siguiente página:



En esta página constan todas las medicinas existentes, además de los botones de administración.

Ingreso de una nueva medicina

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

MEDICINA

⊗ Campo Nombre Medicinal requerido
⊗ Campo Nombre Comercial requerido
⊗ Campo Tipo Presentacion requerido
⊗ Campo Visitador requerido
⊗ Campo Laboratorio requerido
⊗ Campo Stock requerido

INGRESO DE DATOS

Nombre Medicinal

Nombre Comercial

Tipo Presentacion

Visitador

Laboratorio

Stock

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Lapa	Laparamicina	Gotas	JJ	LabComa	10	

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de medicina, mostrando un mensaje de éxito.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

MEDICINA

✔ Registro guardado correctamente

INGRESO DE DATOS

Nombre Medicinal

Nombre Comercial

Tipo Presentacion

Visitador

Laboratorio

Stock

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Lapa	Laparamicina	Gotas	JJ	LabComa	10	
Ampibex	Ampibexina	Gotas	Juan	LabComa	0	

Eliminación de un registro existente de medicina

- Click en el botón Eliminar del registro que se desea eliminar.
- La aplicación mostrará un mensaje de confirmación.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

La página en http://localhost:8080 dice:

Esta seguro de que desea eliminar el registro ?

Cancelar Aceptar

MEDICINA

INGRESO DE DATOS

Nombre Medicinal

Nombre Comercial

Tipo Presentacion

Visitador

Laboratorio

Stock

Nuevo Guardar Cancelar

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Lapa	Laparamicina	Gotas	JJ	LabComa	10	
Ampibex	Ampibexina	Gotas	Juan	LabComa	0	

- Si confirma la eliminación el registro se eliminará, mostrando un mensaje de éxito.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

Registro eliminado correctamente

MEDICINA

INGRESO DE DATOS

Nombre Medicinal

Nombre Comercial

Tipo Presentacion

Visitador

Laboratorio

Stock

Nuevo Guardar Cancelar

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Lapa	Laparamicina	Gotas	JJ	LabComa	10	

Edición de un registro existente

- Click en el botón Editar del registro que se desea editar.
- La aplicación cargará la información del registro seleccionado.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

MEDICINA

INGRESO DE DATOS

Nombre Medicinal

Nombre Comercial

Tipo Presentacion

Visitador

Laboratorio

Stock

Nuevo Guardar Cancelar

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Lapa	Laparamicina	Gotas	JJ	LabComa	10	

- Editar los campos que se deseen.
- Click en el botón guardar para que se actualice el registro seleccionado.
- La aplicación mostrará un mensaje de éxito de guardado.

INVENTARIO MEDICINAS

Administración Control Inventario Reportes

MEDICINA

Registro guardado correctamente

INGRESO DE DATOS

Nombre Medicinal: Laparona

Nombre Comercial: Laparamicina

Tipo Presentacion: Gotas

Visitador: JJ SS

Laboratorio: LabComa

Stock: 10

Nuevo Guardar Cancelar

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Laparona	Laparamicina	Gotas	JJ	LabComa	10	

PAGINA ADMINISTRACION DE EGRESOS DE MEDICINA

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Control Inventario > Egreso.

INVENTARIO MEDICINAS

Administración Control Inventario Reportes

Egreso
Ingreso
Medicina

Al ingresar en esta opción se mostrará la siguiente página:



EGRESO MEDICINAS

Usuario	Fecha Egreso	Acciones
JJSL	2013-12-05 00:00:00.0	<input type="button" value="E"/> <input type="button" value="X"/>
JJSL	2013-12-05 00:00:00.0	<input type="button" value="E"/> <input type="button" value="X"/>

En esta página constan todos los egresos existentes, además de los botones de administración.

Ingreso de un nuevo egreso

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:



EGRESO MEDICINAS

INGRESO DE DATOS

Usuario

Fecha

Medicina	Cantidad	Destino	Descripcion
<input type="text" value="Seleccione..."/>	<input type="text" value="0"/>	<input type="text" value="Seleccione..."/>	<input type="text"/>

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de egreso, mostrando un mensaje de éxito.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

EGRESO MEDICINAS

[Nuevo Egreso](#)

INGRESO DE DATOS

Usuario

Fecha

[Agregar Medicina](#)

Medicina	Cantidad	Destino	Descripcion
Laparamicina	4	Consultorio	Medicamento diario
Seleccione...	0	Seleccione...	

[Guardar](#) [Cancelar](#)



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

EGRESO MEDICINAS

Registro guardado correctamente

[Nuevo Egreso](#)

Usuario	Fecha Egreso	Acciones
JJSL	2013-12-05 00:00:00.0	
JJSL	2013-12-05 00:00:00.0	
JJSL	2014-01-13 00:00:00.0	
JJSL	2014-01-13 00:00:00.0	

- Después de que se haya guardado el egreso se puede verificar que el stock de la medicina seleccionada disminuyo:

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Laparona	Laparamicina	Gotas	JJ	LabComa	6	

Ver egreso

- Click en el botón ver de los egresos
- La aplicación cargará la información del egreso seleccionado



Administración Control Inventario Reportes

EGRESO MEDICINAS

Nuevo Egreso

INGRESO DE DATOS

Usuario: JJSL

Fecha: ene 13, 2014

Agregar Medicina

Medicina	Cantidad	Destino	Descripcion
Laparamicina	4	Consultorio	Medicamento diario

Guardar Cancelar

PAGINA ADMINISTRACION DE INGRESOS DE MEDICINA

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Control Inventario > Ingreso.



Administración Control Inventario Reportes

- Egreso
- Ingreso
- Medicina

Al ingresar en esta opción se mostrará la siguiente página:



Administración Control Inventario Reportes

INGRESO MEDICINAS

Nuevo Ingreso

Usuario	Fecha Ingreso	Acciones
JJSL	2013-12-04 00:00:00.0	[icon] [X]
JJSL	2013-12-05 00:00:00.0	[icon] [X]

En esta página constan todos los ingresos existentes, además de los botones de administración.

Ingreso de un nuevo ingreso de medicina

- Click en el botón de nuevo, esto habilitara los campos para ingresar los datos del registro a crear.
- Ingresar todos los campos requeridos, caso contrario la aplicación no le permitirá almacenar dando el siguiente error:



The screenshot shows the 'INVENTARIO MEDICINAS' application interface. At the top left is the 'SB' logo. The navigation bar includes 'Administración', 'Control Inventario', and 'Reportes'. The main window is titled 'EGRESO MEDICINAS' and contains a 'Nuevo Egreso' button. Below it, the 'INGRESO DE DATOS' section has fields for 'Usuario' (a dropdown menu) and 'Fecha' (a date picker). An 'Agregar Medicina' button is located below the form. A table with the following structure is displayed:

Medicina	Cantidad	Destino	Descripcion
Seleccione...	0	Seleccione...	

At the bottom of the form are 'Guardar' and 'Cancelar' buttons. A red error message is visible at the top of the form area, listing four required fields: 'Campo Usuario requerido', 'Campo Fecha requerido', 'Campo Medicina requerido', and 'Campo Destino requerido'.

- Si se ha ingresado todos los campos requeridos dar click en el botón guardar.
- Al dar click en dicho botón se creara un nuevo registro de ingreso, mostrando un mensaje de éxito.



The screenshot shows the 'INVENTARIO MEDICINAS' application interface after a successful entry. The 'INGRESO DE DATOS' section now has 'Usuario' set to 'JJSL' and 'Fecha' set to 'ene 13, 2014'. The 'Agregar Medicina' button is still present. The table now contains two rows:

Medicina	Cantidad	Fecha Caducidad	Descripcion
Laparamicina	3	ene 7, 2014	Ingreso
Seleccione...	0	ene 13, 2014	

The 'Guardar' and 'Cancelar' buttons remain at the bottom of the form.



Administración Control Inventario Reportes

INGRESO MEDICINAS
Registro guardado correctamente
Nuevo Ingreso

Usuario	Fecha Ingreso	Acciones
JJSL	2013-12-04 00:00:00.0	
JJSL	2013-12-05 00:00:00.0	
JJSL	2014-01-13 00:00:00.0	

- Después de que se haya guardado el egreso se puede verificar que el stock de la medicina seleccionada se incrementó:

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock	Acciones
Laparona	Laparamicina	Gotas	JJ	LabComa	9	

Ver egreso

- Click en el botón ver de los ingresos
- La aplicación cargará la información del ingreso seleccionado



Administración Control Inventario Reportes

INGRESO MEDICINAS
Nuevo Ingreso

INGRESO DE DATOS
Usuario: JJSL
Fecha: ene 13, 2014

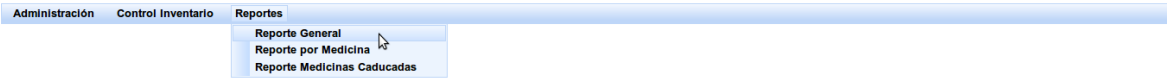
Agregar Medicina

Medicina	Cantidad	Fecha Caducidad	Descripcion
Laparamicina	3	ene 7, 2014	Ingreso

Guardar Cancelar

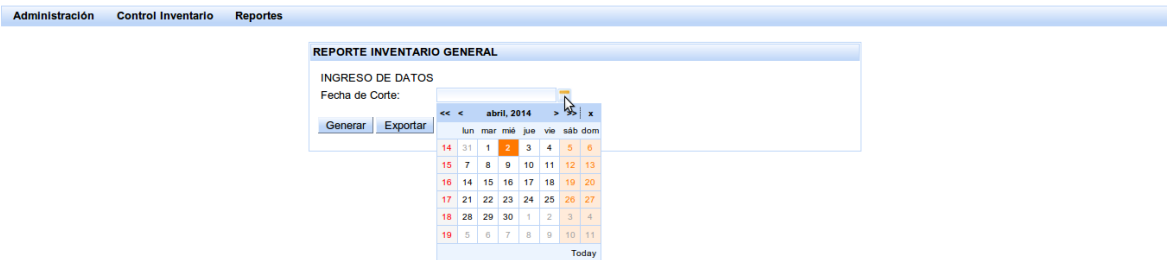
PAGINA REPORTE INVENTARIO GENERAL

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Reportes > Reporte General.

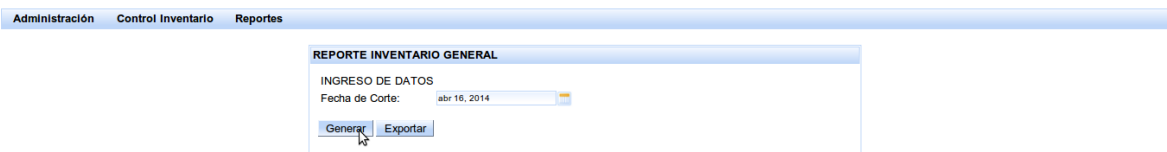


Generación de Reporte General del Inventario de Medicinas

- Primero si se desea se puede escoger una fecha corte hasta la cual se desea generar el reporte general del inventario.



- A continuación Click en el botón Generar, el cual generará el reporte para la fecha seleccionada.





REPORTE INVENTARIO GENERAL

INGRESO DE DATOS
Fecha de Corte:

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock
Laparona	Laparamicina	Gotas	JJ	LabComa	9
Tirotricina	Bucotricina	Gotas	Juan	LabComa	3
Peroxido de Hidrogeno	Agua Oxigenada	Gotas	JJ	LabComa	4
Hidroxido de Magnesio	Magnezia	Gotas	Juan	LabComa	5

PAGINA REPORTE INVENTARIO POR MEDICINA

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Reportes > Reporte por Medicina.

Generación de reporte Inventario por Medicina.

- Primero si se desea se puede escoger una fecha de corte hasta la cual se desea generar el reporte.



REPORTE INVENTARIO POR MEDICINA

INGRESO DE DATOS
Fecha de Corte:

Medicina:

- A continuación seleccionar la medicina de la cual se desea el reporte desde el combo de medicinas.



REPORTE INVENTARIO POR MEDICINA

INGRESO DE DATOS
Fecha de Corte:

Medicina:

- Click en el botón Generar, el cual generará el reporte para la fecha seleccionada y de la medicina seleccionada.

INVENTARIO MEDICINAS

Administración Control Inventario Reportes

REPORTE INVENTARIO POR MEDICINA

INGRESO DE DATOS

Fecha de Corte: abr 18, 2014

Medicina: Bucotricina

Generar Exportar

INVENTARIO MEDICINAS

Administración Control Inventario Reportes

REPORTE INVENTARIO GENERAL

INGRESO DE DATOS

Fecha de Corte: abr 16, 2014

Generar Exportar

Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Visitador	Laboratorio	Stock
Laparon	Laparamicina	Gotas	JJ	LabComa	9
Tirotricina	Bucotricina	Gotas	Juan	LabComa	3
Peroxido de Hidrogeno	Agua Oxigenada	Gotas	JJ	LabComa	4
Hidroxido de Magnesio	Magnezia	Gotas	Juan	LabComa	5

PAGINA REPORTE INVENTARIO MEDICINA CADUCADA

Para ingresar a esta página se debe acceder desde el menú de la aplicación accediendo a la opción Reportes > Reporte Medicinas Caducadas.

INVENTARIO MEDICINAS

Administración Control Inventario Reportes

- Reporte General
- Reporte por Medicina
- Reporte Medicinas Caducadas

Generación de Reporte Medicinas Caducadas

- Click en el botón Generar, el cual generará el reporte de todas las medicinas caducas.



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

REPORTE MEDICINAS CADUCADAS

Generar Exportar



INVENTARIO MEDICINAS

Administración Control Inventario Reportes

REPORTE MEDICINAS CADUCADAS

Generar Exportar

Lote	Nombre Medicinal	Nombre Comercial	Tipo Presentacion	Cantidad	Fecha Caducidad	Costo \$
1	Laparon	Laparamicina	Gotas	10	2013-12-04 00:00:00.0	65
51	Laparon	Laparamicina	Gotas	2	2013-12-05 00:00:00.0	72
52	Laparon	Laparamicina	Gotas	8	2013-12-05 00:00:00.0	16
151	Laparon	Laparamicina	Gotas	3	2014-01-07 00:00:00.0	75

“APLICACIÓN DISTRIBUIDA BASADO EN LAS
ESPECIFICACIONES JEE6 PARA EL CONTROL DEL
INVENTARIO DE MEDICINAS EN LA CLÍNICA SAN
BARTOLOMÉ”

OBJETIVO: EL OBJETIVO DE ESTE MANUAL ES DE
PROPORCIONAR UNA GUÍA AL USUARIO TÉCNICO DE LA
APLICACIÓN PARA QUE PUEDA ADMINISTRAR EL
SERVIDOR DE APLICACIONES

Requerimientos de Software

Para que la aplicación de Control de Inventario de Medicinas pueda ser ejecutada correctamente, se debe tener instalados los siguientes requerimientos de software:

- Ambiente de desarrollo de aplicaciones Java: Java SE Development Kit 6u30 (Java JDK version 1.6)
- Servidor de aplicaciones para desplegar la aplicación: JBoss Application Server 7.1.1.Final
- Seleccionar el link para descargar el zip o tar.gz de la version 7.1.1 Final (Everything (NOT Java EE6 Certified))
- Motor de base de datos: PostgreSQL 9.1.2-1
- Herramienta de administración de proyectos: Maven 3.0.4

Configuración del servidor de aplicaciones

Una vez descargado el archivo zip del servidor de aplicaciones lo descomprimos.

Ya descomprimido el archivo obtendremos una carpeta con el nombre:

“Jboss-as-7.1.1.final” que es el servidor de aplicaciones.

Se debe configurar en el servidor JBOSS lo siguiente:

Configuración del Datasource

Se debe agregar el origen de datos a donde se conectara nuestra aplicación dentro del tag `<datasources></datasources>` en el archivo `/standalone/configuration/standalone.xml` del servidor de aplicaciones.

Editar la dirección IP, puerto, nombre de usuario y contraseña de acuerdo a la instalación del motor de base de datos PostgreSQL correspondiente.

Además agregar el driver correspondiente a PostgreSQL dentro del tag `<drivers></drivers>`.

```
standalone.xml
<subsystem xmlns="urn:jboss:domain:datasources:1.0">
  <datasources>
    <datasource jndi-name="java:jboss/datasources/ExampleDS" pool-name="ExampleDS" enabled="true" use-java-context="true">
      <connection-url>jdbc:h2:mem:test;DB_CLOSE_DELAY=-1</connection-url>
      <driver>h2</driver>
      <security>
        <user-name>sa</user-name>
        <password>sa</password>
      </security>
    </datasource>
    <datasource jta="true" jndi-name="java:jboss/datasources/inventarioDS" pool-name="inventarioDS_pool" enabled="true" use-java-context="true" use-
ccm="true">
      <connection-url>jdbc:postgresql://localhost:5432/inventario</connection-url>
      <driver>postgresql</driver>
      <pool>
        <min-pool-size>10</min-pool-size>
        <max-pool-size>100</max-pool-size>
        <prefill>true</prefill>
        <use-strict-min>false</use-strict-min>
        <flush-strategy>FailingConnectionOnly</flush-strategy>
      </pool>
      <security>
        <user-name>postgres</user-name>
        <password>postgres</password>
      </security>
    </datasource>
    <drivers>
      <driver name="h2" module="com.h2database.h2">
        <xa-datasource-class>org.h2.jdbcx.JdbcDataSource</xa-datasource-class>
      </driver>
      <driver name="postgresql" module="org.postgresql">
        <xa-datasource-class>org.postgresql.xa.PGXADatasource</xa-datasource-class>
      </driver>
    </drivers>
  </datasources>

```

Ejecución del servidor de aplicaciones

Para levantar la aplicación que se encuentra dentro del servidor de aplicaciones se debe realizar lo siguiente:

1. Abrir una terminal de comandos.
2. En el terminal de comandos acceder a la dirección donde se encuentra el servidor de aplicaciones JBoss con el comando “cd”.
3. Una vez que se encuentre en el directorio del servidor de aplicaciones acceder al siguiente path:

```
cd Jboss-as-7.1.1.final\standalone\bin
```

4. A continuación ejecutar el siguiente comando con el cual se levantará el servidor de aplicaciones con la aplicación para el control del inventario:

```
./standalone.sh
```

5. Con esto ya se puede acceder a la aplicación