



UNIVERSIDAD DE GUAYAQUIL

**FACULTAD DE CIENCIAS MATEMATICAS Y FISICAS
CARRERA DE INGENIERIA EN SISTEMAS
COMPUTACIONALES**

**DESARROLLO DEL FORMULARIO ANEXO1-FICHA FAMILIAR SECCIÓN1
DEL MINISTERIO DE SALUD PÚBLICA APLICANDO ARQUETIPOS
BASADO EN NORMA ISO 13606 PARA OBTENER
INTEROPERABILIDAD ENTRE SISTEMAS
HOSPITALARIOS**

MANUAL DE USUARIO

AUTORES:

**CARLOS ANTONIO MEDINA BRAVO
FERNANDO RICARDO FLORES MOLINA**

**GUAYAQUIL – ECUADOR
SEPTIEMBRE - 2016**

ÍNDICE DE GRÁFICOS

GRÁFICO 1: Pantalla de Login.....	2
GRÁFICO 2: Pantalla Principal.	3
GRÁFICO 3: Formulario Ficha Familiar/ Datos de la Vivienda.....	4
GRÁFICO 4: Formulario Ficha Familiar/ Mortalidad Familiar.....	6
GRÁFICO 5: Formulario Ficha Familiar/ Evaluación Familiar.....	7
GRÁFICO 6: Formulario Ficha Familiar/ Diagnostico Familiar.....	8

INTRODUCCIÓN

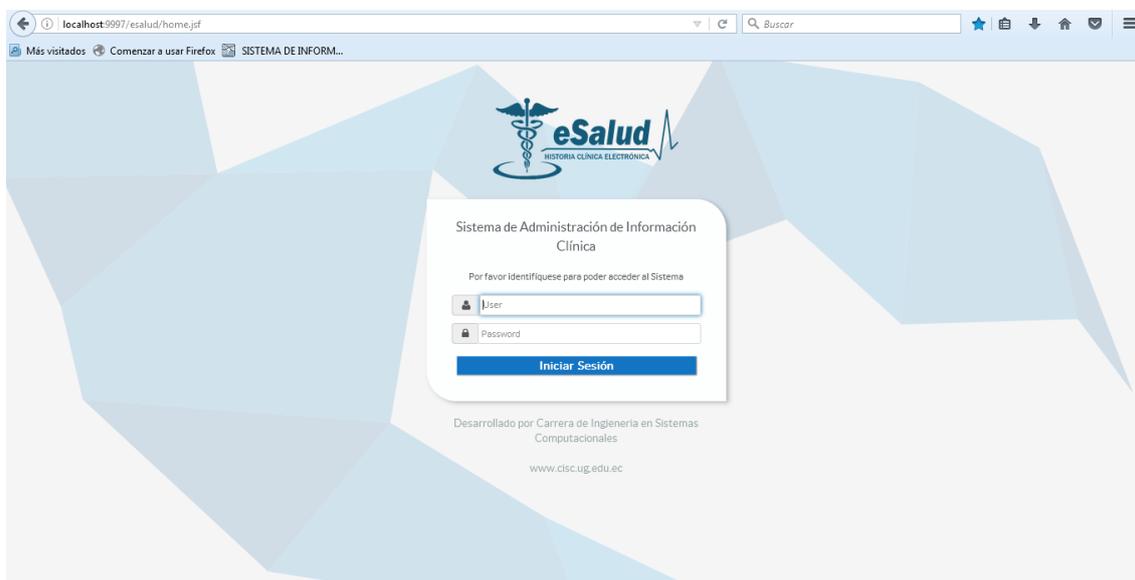
El presente documento tiene como objetivo ser de ayuda para el usuario sobre el uso y manejo del formulario Ficha Familiar sección uno.

El detalle de ingreso y uso del formulario se explica a continuación.

INGRESO AL SISTEMA

Se deberá ingresar la siguiente ruta en el navegador para acceder a la aplicación:

GRÁFICO 1: Pantalla de Login

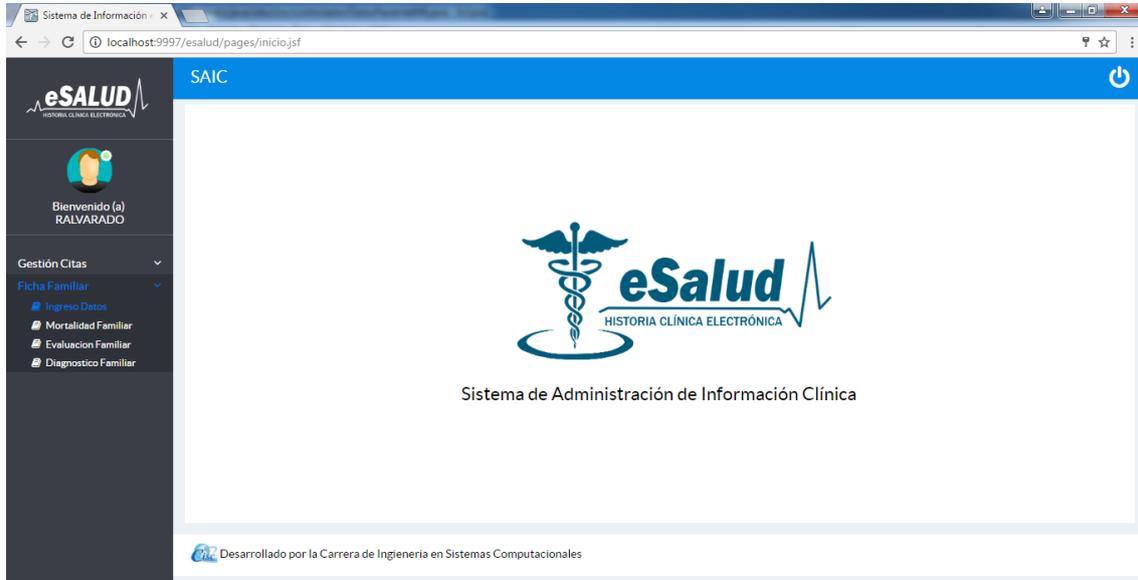


Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

En esta pantalla se ingresara el usuario y contraseña, estos campos serán validados por la aplicación, si son válidos se permitirá ingresar a la opción principal.

GRÁFICO 2: Pantalla Principal



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Esta pantalla principal, tendrá un menú con las opciones de los formularios: Lista de Problemas, Notas de Evolución, Prescripciones Médicas, Lista de Diagnósticos y Manejo, **Concentrado de Laboratorio**, y debemos dar clic en cualquiera de las opciones para que nos dirija al formulario indicado en las opciones del menú.

GRÁFICO 3: Formulario Ficha Familiar/ Datos de la Vivienda.

Gestión Usuario

Buscar por identificación

Datos de la Vivienda

Identificación	<input type="text"/>	Primer Nombre	<input type="text"/>
Segundo Nombre	<input type="text"/>	Apellido Paterno	<input type="text"/>
Apellido Materno	<input type="text"/>	Identificador del Establecimiento	<input type="text"/>
Provincia	<input type="text"/>	Canton	<input type="text"/>
Parroquia	<input type="text"/>	Distrito	<input type="text"/>
Zona	<input type="text"/>	Localidad	<input type="text"/>
Sector	<input type="text"/>		

Direccion Vivienda

Manzana	<input type="text"/>	Division	<input type="text"/>
Edificio	<input type="text"/>	Numero de Vivienda	<input type="text"/>
Numero de Hogar	<input type="text"/>	Calle Principal	<input type="text"/>
Calle Secundaria	<input type="text"/>	Numero de Casa	<input type="text"/>
Conjunto Habitacional	<input type="text"/>	Numero de Lote	<input type="text"/>

Gestión Usuario

Condicion Ocupacion de la Vivienda

Estado de la Vivienda

Datos de la Vivienda

Tipo de Vivienda	<input type="text"/>	Acceso Principal a la Vivienda	<input type="text"/>
Techo de la Vivienda	<input type="text"/>	Piso de la Vivienda	<input type="text"/>
Paredes de la Vivienda	<input type="text"/>	Estado del Techo de la Vivienda	<input type="text"/>
Estado del Piso de la Vivienda	<input type="text"/>	Estado de la Pared de la Vivienda	<input type="text"/>

Datos de Hogar de la Vivienda

Estado de ocupacion de la Vivienda	<input type="text"/>	Cantidad de Cuartos en el Hogar	<input type="text"/>
Cantidad de Dormitorios en el Hogar	<input type="text"/>	Procedencia del Agua en el Hogar	<input type="text"/>
De donde se recibe el agua	<input type="text"/>	Tratamiento del agua para beber	<input type="text"/>
Tipo de Servicio Higienico que cuenta en el Hogar	<input type="text"/>	Ubicacion del Servicio Higienico	<input type="text"/>
Servicio de ducha que tiene en el hogar	<input type="text"/>	Eliminacion de la Rescura	<input type="text"/>

The screenshot shows a web browser window with the URL `localhost:9997/esalud/pages/ficha_familiar/datosPaciente.jsf`. The page title is "Gestión Usuario". The interface is in Spanish and includes a sidebar with the following navigation items: "Gestión Citas", "Ficha Familiar", "Ingreso Datos", "Mortalidad Familiar", "Evaluación Familiar", and "Diagnóstico Familiar". The main content area contains a form with the following sections:

- Servicio de ducha que tiene en el hogar:** A dropdown menu.
- Tipo de Alumbrado:** A dropdown menu.
- Tipo de Transporte:** A dropdown menu.
- Eliminación de la Basura:** A dropdown menu.
- Tipo de Cocina:** A dropdown menu.
- Tiempo de Desplazamiento:** A dropdown menu.
- Factores de Vulnerabilidad de la Vivienda:**
 - Riesgos Medioambientales:** A dropdown menu.
 - Riesgos Sanitarios:** A dropdown menu.
 - Riesgos de Animales:** A dropdown menu.
- Control:**
 - Control de Entrevistas Efectivas:** A dropdown menu.
 - Control de Trabajo: Código del Encuestador:** A text input field.
 - Numero de Visita:** A text input field.
 - Control de Trabajo: Nombre del Encuestador:** A text input field.
 - Fecha de ultima visita:** A text input field.

At the bottom of the form is a blue "Guardar" button. The footer of the page reads "Desarrollado por la Carrera de Ingeniería en Sistemas Computacionales".

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

En este formulario se ingresará la información del formulario ficha familiar datos de la vivienda de una persona que en este caso sería el jefe del hogar, o en su defecto la persona que nos atienda en ese momento en el hogar. Tenemos los datos que se debe ingresar como identificación, nombres, apellidos, provincia, cantón, localidad, etc.

A su vez se debe llenar la dirección de la vivienda como nombres de calles, numeración de la vivienda, número de lote, numero de edificio, nombre de conjunto habitacional, etc.

Este formulario también es necesario llenar si cuando se realiza la visita domiciliaria estaba habitada la vivienda para poder continuar con los demás puntos.

Además, tenemos los datos de la vivienda exacto como tipo de piso, tipo de techo, tipo de pared y los estados de los mismos.

Finalmente encontramos los factores de riesgos que puede tener el hogar como la contaminación, los animales y riesgos sanitarios; con los respectivos controles de la ficha familiar que encontramos responsable de llenar la ficha familiar, fecha, código de responsable y numero de visita domiciliaria.

GRÁFICO 4: Formulario Ficha Familiar/ Mortalidad Familiar.

The image shows a web browser window displaying a form titled 'Mortalidad Familiar' within a system called 'eSALUD'. The browser address bar shows 'localhost:9997/esalud/pages/ficha_familiar/mortalidadFamiliar.jsf'. The page has a blue header with 'Gestión Usuario' and a search bar with 'Buscar por identificación' and buttons for 'Consultar' and 'Limpiar'. The form itself is titled 'Mortalidad Familiar' and contains the following fields:

- Nombre de Familiar
- Apellido de Familiar
- Sexo de Familiar
- Edad de Familiar
- Documentacion de Familiar
- Identificacion Familiar
- Parentesco Familiar (Con relacion al jefe del hogar)
- Fecha Muerte de Familiar
- Clase de muerte del Familiar
- Causa de la Muerte de Familiar
- Observacion

At the bottom of the form is a 'Guardar' button. The footer of the page indicates it was developed by the 'Carrera de Ingeniería en Sistemas Computacionales'.

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

En este formulario se ingresará la información del formulario mortalidad familiar en el cual se ingresa y se busca los familiares que han fallecido dentro de los últimos cinco años, los datos que se deben llenar son los nombres, apellidos sexo, edad de familiar si es menor a cinco años con meses, identificación, parentesco familiar, fecha de muerte del familiar, clase de muerte, causa de la muerte y una observación.

GRÁFICO 5: Formulario Ficha Familiar/ Evaluación Familiar.

The screenshot shows a web browser window with the URL `localhost:9997/esalud/pages/ficha_familiar/intervencionesFamiliar.jsf`. The page title is "Gestión Usuario". On the left, there is a sidebar with the "eSALUD" logo and a navigation menu including "Bienvenido (a) RALVARADO", "Gestión Citas", and "Ficha Familiar" (with sub-items: "Ingreso Datos", "Mortalidad Familiar", "Evaluación Familiar", and "Diagnostico Familiar"). The main content area is titled "Evaluación de las Intervenciones Familiar (Registre de acuerdo a los codigos)". It contains a search bar with "Buscar por identificación" and buttons for "Consultar" and "Limpiar". Below this is a form with the following fields: "Identificación", "Fecha de Registro", "Codigo de Resultados", "Detalle de los Acuerdos alcanzados", "Nombre Completo", "Codigo Cumplimiento", and "Causas y Dificultades". A "Guardar" button is located at the bottom of the form. At the bottom of the page, it says "Desarrollado por la Carrera de Ingeniería en Sistemas Computacionales".

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

En este formulario se ingresará la información del formulario evaluación familiar en el cual se ingresa y se busca los seguimientos de las evaluaciones de las intervenciones que los familiares han sufrido; los datos que se deben de llenar son identificación, nombre completo, fecha de ingreso, código cumplimiento, código de resultados, causas y dificultades y detalle de los acuerdos alcanzados.

GRÁFICO 6: Formulario Ficha Familiar/ Diagnostico Familiar.

The image displays two screenshots of a web application interface for 'Gestión Usuario'. The top screenshot shows the 'Actividad de Diagnostico Familiar' section with three diagnostic visit forms. The bottom screenshot shows a zoomed-in view of the 'Visita de Segundo Diagnostico' form, highlighting the input fields for 'Clasificacion Diagnostica', 'Describir el Problema Actual', 'Nombre Completo', 'Fecha de Diagnostico', and 'Nombre Responsable'. The interface includes a search bar, a sidebar with navigation options, and a footer indicating it was developed by the Carrera de Ingeniería en Sistemas Computacionales.

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

En este formulario se ingresará la información del formulario diagnostico familiar en el cual se ingresa y se busca las diferentes visitas de diagnóstico; la información; identificación, nombre completo, clase de diagnóstico, fecha de diagnóstico, describir el problema actual, nombre de responsable.



UNIVERSIDAD DE GUAYAQUIL

**FACULTAD DE CIENCIAS MATEMATICAS Y FISICAS
CARRERA DE INGENIERIA EN SISTEMAS
COMPUTACIONALES**

**“DESARROLLO DEL FORMULARIO ANEXO1-FICHA FAMILIAR SECCIÓN1
DEL MINISTERIO DE SALUD PÚBLICA APLICANDO ARQUETIPOS
BASADO EN NORMA ISO 13606 PARA OBTENER
INTEROPERABILIDAD ENTRE SISTEMAS
HOSPITALARIOS”**

MANUAL TÉCNICO

AUTORES:

**CARLOS ANTONIO MEDINA BRAVO
FERNANDO RICARDO FLORES MOLINA**

**GUAYAQUIL – ECUADOR
SEPTIEMBRE - 2016**

INDICE DE GRÁFICOS

GRÁFICO 1: Paso 1-Instalación de Java	4
GRÁFICO 2: Paso 2-Instalación de Java	6
GRÁFICO 3: Paso 3-Instalación de Java	6
GRÁFICO 4: Paso 4-Instalación de Java	7
GRÁFICO 5: Paso 5-Instalación de Java	7
GRÁFICO 6: Paso 1-Instalación de Eclipse Neón	8
GRÁFICO 7: Paso 2-Instalación de Eclipse Neón	9
GRÁFICO 8: Paso 3-Instalación de Eclipse Neón	9
GRÁFICO 9:Paso1-Instalación de Maven	10
GRÁFICO 10:Paso2-Instalación de Maven	10
GRÁFICO 11:Paso1-Creación de un Proyecto Spring MVC	11
GRÁFICO 12:Paso2-Creación de un Proyecto Spring MVC	11
GRÁFICO 13:Paso3-Creación de un Proyecto Spring MVC	12
GRÁFICO 14:Paso1-Instalación de Spring	13
GRÁFICO 15:Paso2-Instalación de Spring	13
GRÁFICO 16:Paso3-Instalación de Spring	14
GRÁFICO 17:Paso1- Instalación de Jboss Tools 4.4.0.Final	15
GRÁFICO 18:Paso2- Instalación de Jboss Tools 4.4.0.Final	15
GRÁFICO 19: Instalación del Servidor Widfly	16
GRÁFICO 20: Paso1-Configurar Variable Java Home	17
GRÁFICO 21: Paso2-Configurar Variable Java Home	18
GRÁFICO 22: Paso3-Configurar Variable Java Home	19
GRÁFICO 23: Configurar Variable Path	20
GRÁFICO 24: Paso1- Configurar Maven en Windows	21
GRÁFICO 25: Paso2- Configurar Maven en Windows	22
GRÁFICO 26: Paso3- Configurar Maven en Windows	23
GRÁFICO 27: Paso4- Configurar Maven en Windows.....	23
GRÁFICO 28: Paso1- Instalación de PostgreSQL	24
GRÁFICO 29: Paso2- Instalación de PostgreSQL	25
GRÁFICO 30: Paso3- Instalación de PostgreSQL	25
GRÁFICO 31: Paso4- Instalación de PostgreSQL	26
GRÁFICO 32: Paso5- Instalación de PostgreSQL	26
GRÁFICO 33: Paso6- Instalación de PostgreSQL	27

GRÁFICO 34: Paso7- Instalación de PostgreSQL	27
GRÁFICO 35: Paso8- Instalación de PostgreSQL	28
GRÁFICO 36: Paso9- Instalación de PostgreSQL	28
GRÁFICO 37: Paso10- Instalación de PostgreSQL	29
GRÁFICO 38: Conexión a la Base de Datos	29
GRÁFICO 39: Creación de Nuevo Rol	30
GRÁFICO 40: Creación de la Base de Datos	30
GRÁFICO 41:Paso1- Importar Base de Datos	31
GRÁFICO 42:Paso2- Importar Base de Datos	31

INTRODUCCIÓN

Se elaboró este documento para que sirva de guía, para el desarrollo de futuros proyectos.

A continuación se describen los requerimientos de hardware y software que se deben utilizar para desarrollar el proyecto.

Requerimientos de Hardware

Se cuenta con los siguientes Equipos:

Laptop:

- Fabricante : LENOVO
- Modelo :Ideapad
- Procesador : Intel(R) Core(TM) i7-4700MQ CPU 2.40GHZ
- Memoria RAM: 12.0 GB.
- Disco duro : 1 TB

Requerimientos de Software

El Sistema operativo que se pueda utilizar para la implementación del Proyecto

Para la instalación de la base de datos PostgreSQL.

- **Sistema Operativo:** Windows 7 o superior en sus diferentes versiones de 32 y 64 bits

Para la instalación de las herramientas de desarrollo.

- **Sistema Operativo:** Windows 7 o superior en su diferentes versiones de 32 y 64 bits

Herramientas de Desarrollo

Se detallan las Herramientas de desarrollo a utilizar:

- IDE de desarrollo : Eclipse Neon 64 bits
- Capa de Presentación : JSF + Primefaces
- Capa de Persistencia : JPA 2.1 + Hibernate
- Framework: Spring MVC

- Servidor de Aplicaciones: Wildfly 10.0
- Manejadores de Bases de datos: PostgreSQL 9.5.3
- Lenguaje de Programación: Java
- Plataforma de Software para el modelado , la normalización y la interoperabilidad semántica de los datos: LinkEHR
- JDK 1.8
- Apache Maven 3.3.9

Instalación de Java JDK

Para instalar Java se deben seguir los siguientes pasos:

- 1.- Ir a la página Oficial de Oracle para descargar la versión de JDK 1.8:
<http://www.oracle.com/technetwork/articles/javase/index-jsp-138363.html>
- 2.-Escoger la versión del sistema operativo donde se va a instalar
- 3.- Una vez descargado ejecutar la instalación.
- 4.-Se siguen los siguientes pasos para la instalación

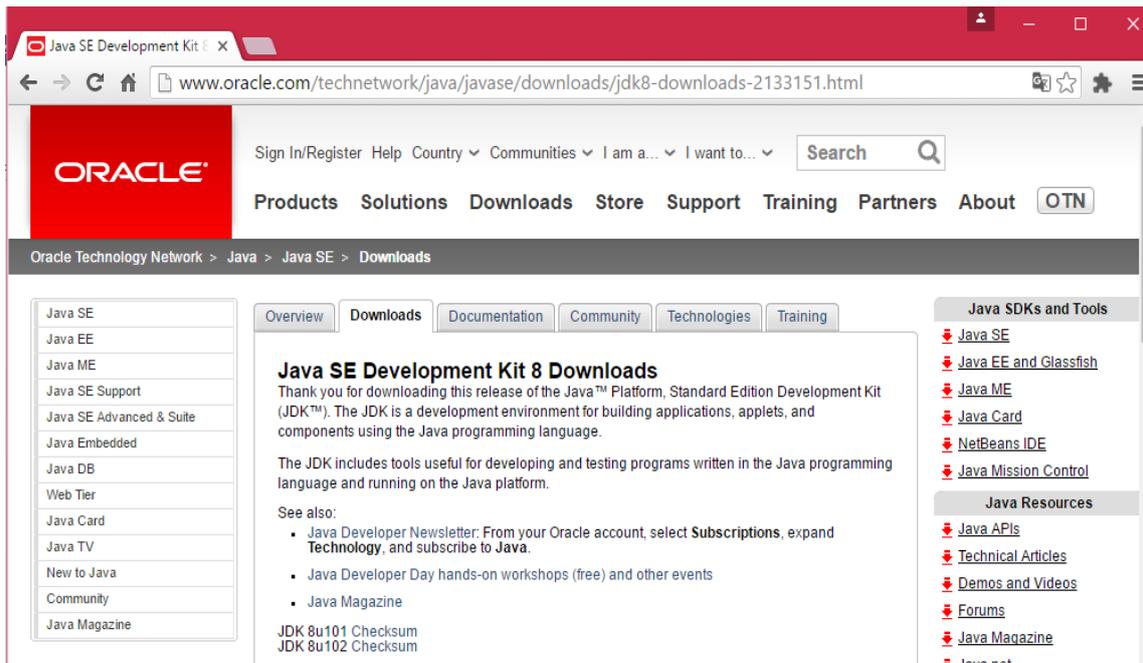
GRÁFICO 1: Paso 1-Instalación de Java



Elaborado por : Carlos Medina, Fernando Flores

Fuente: <http://www.oracle.com/technetwork/articles/javase/index-jsp-138363.html>

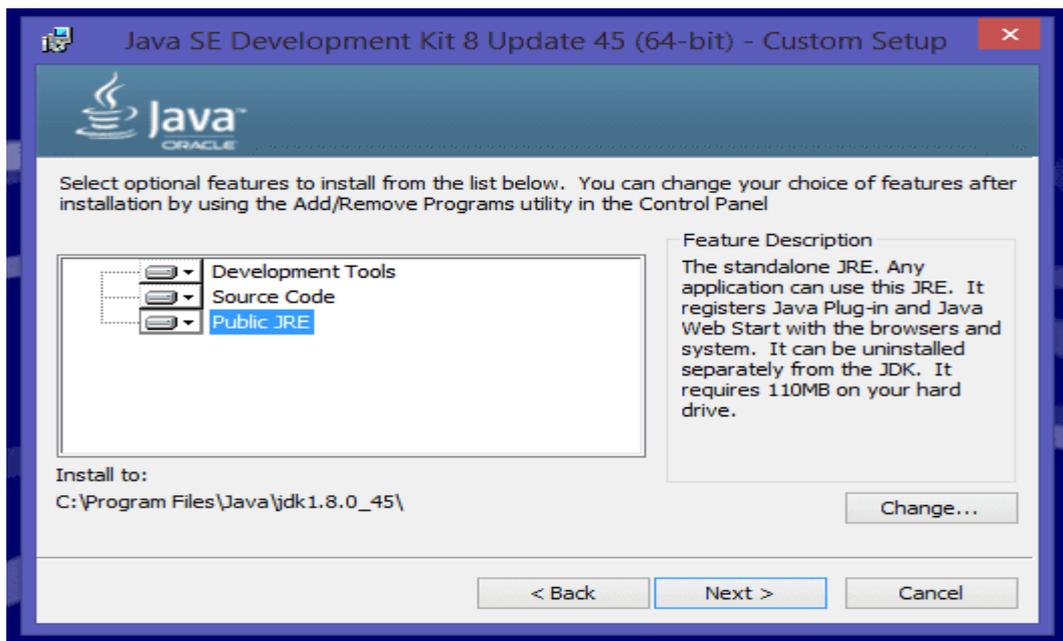
GRÁFICO 2: Paso 2-Instalación de Java



Elaborado por : Carlos Medina, Fernando Flores

Fuente: <http://www.oracle.com/technetwork/articles/javase/index-jsp-138363.html>

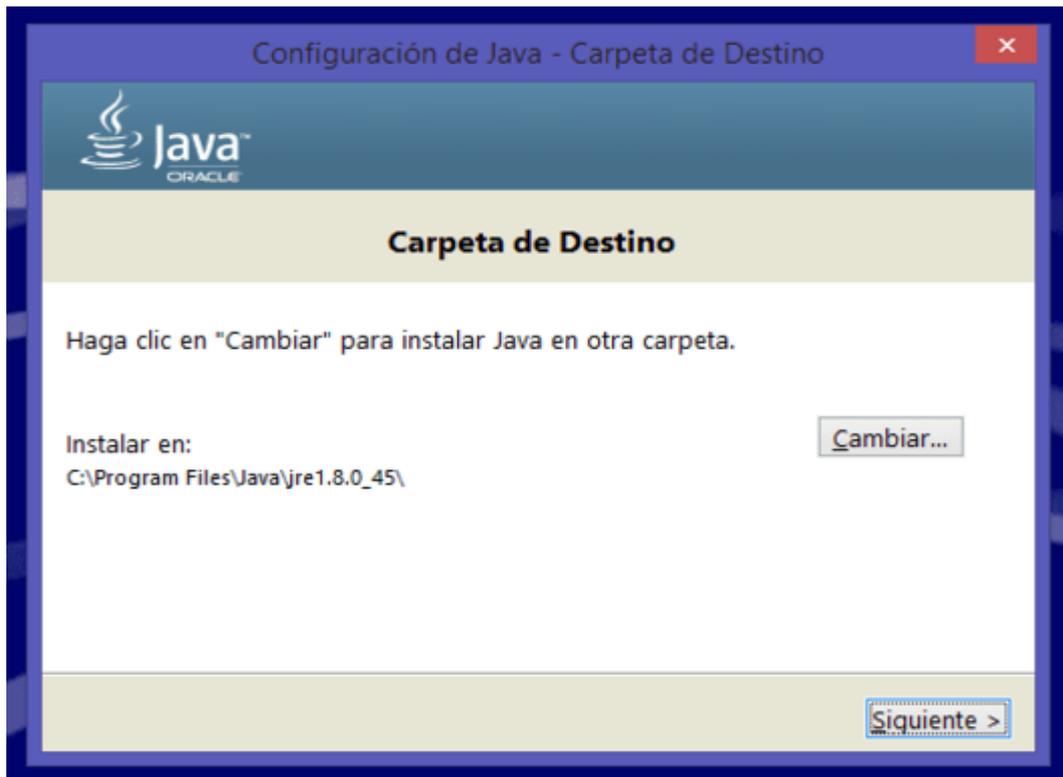
GRÁFICO 3: Paso 3-Instalación de Java



Elaborado por : Carlos Medina, Fernando Flores

Fuente Datos de Investigación

GRÁFICO 4: Paso 4-Instalación de Java



Elaborado por : Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

GRÁFICO 5: Paso 5-Instalación de Java



Elaborado por : Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Instalación de Eclipse Neón

Para instalar Eclipse Neón se siguen los siguientes pasos:

1.-Nos dirigimos a la página oficial para descarga el software:

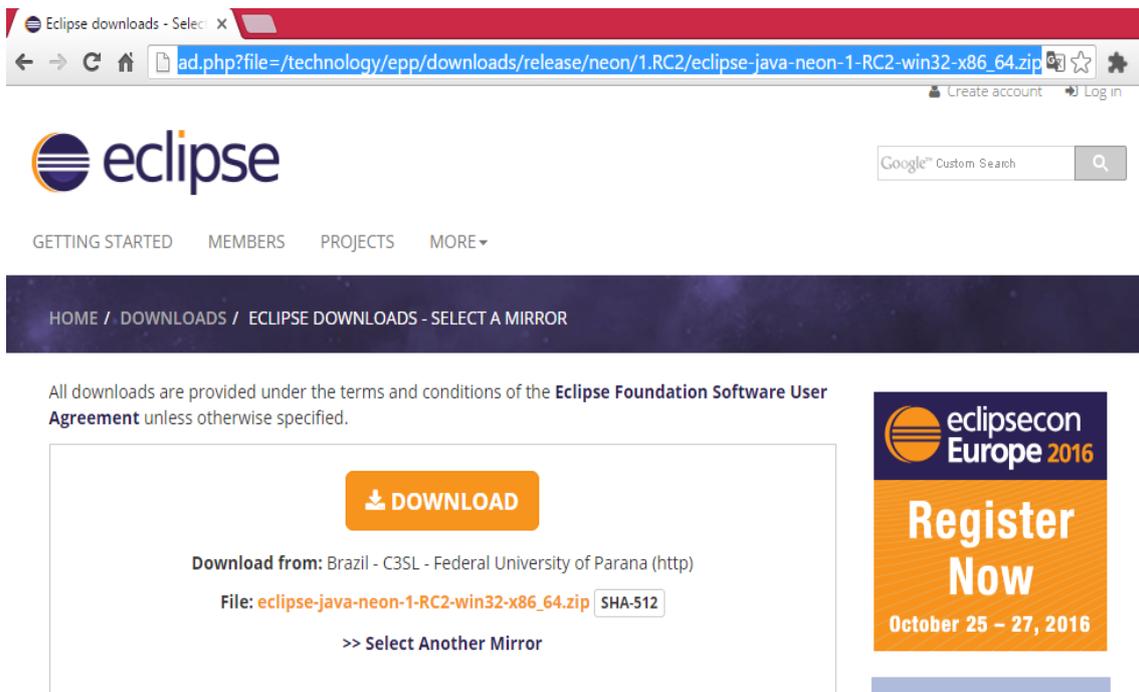
http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/neon/1.RC2/eclipse-java-neon-1-RC2-win32-x86_64.zip

2.-Escogemos la versión para nuestro sistema operativo y seleccionamos descargar.

3. Nos ubicamos en la ruta donde se descargó el programa y le damos ejecutar

4.- Configuramos la ruta donde queremos que se instale el programa.

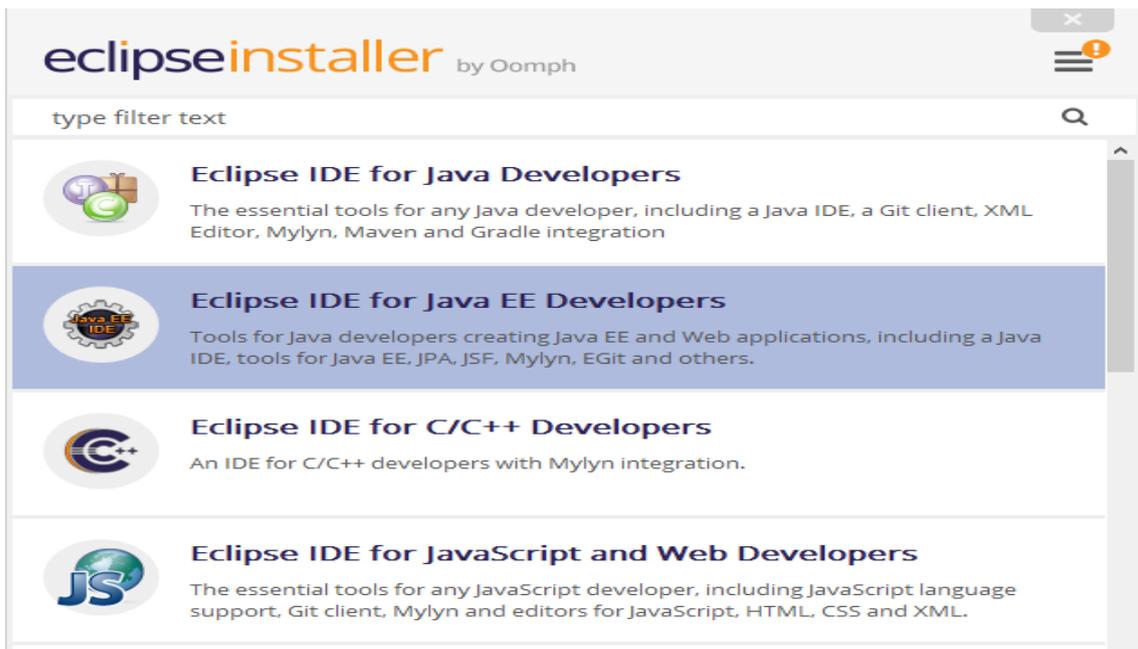
GRÁFICO 6: Paso 1-Instalación de Eclipse Neón



Elaborado por : Carlos Medina, Fernando Flores

Fuente:http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/neon/1.RC2/eclipse-java-neon-1-RC2-win32-x86_64.zip

GRÁFICO 7: Paso 2-Instalación de Eclipse Neón



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

GRÁFICO 8: Paso 3-Instalación de Eclipse Neón



Elaborado por : Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Instalación de Maven

Para instalar Maven se siguen los siguientes pasos:

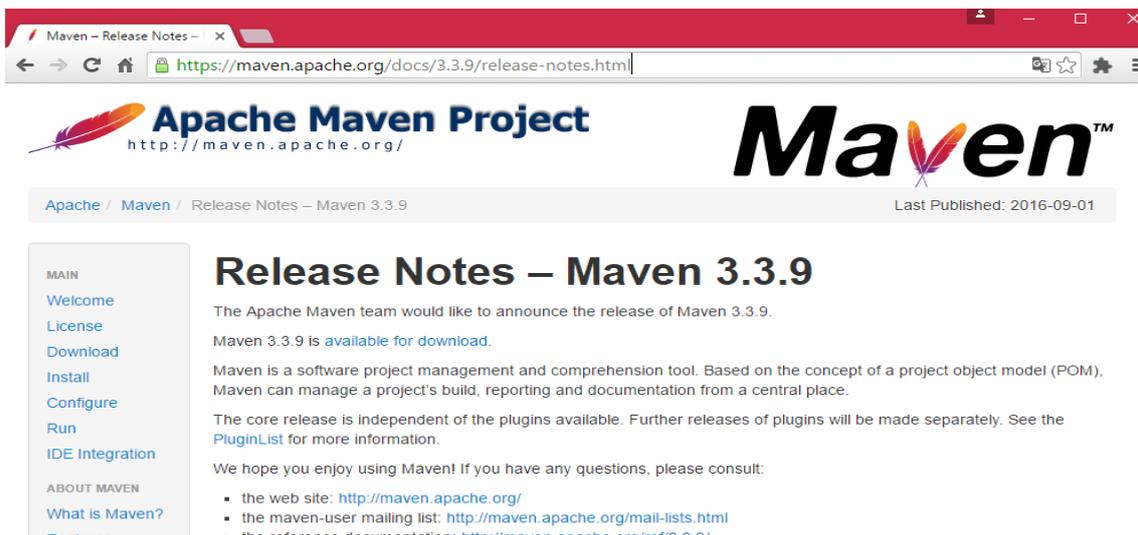
1.-Nos dirigimos a la página oficial para descarga el software:

<https://maven.apache.org/docs/3.3.9/release-notes.html>

2.-Escogemos la versión para nuestro sistema operativo y seleccionamos descargar.

3. Nos ubicamos en la ruta donde se descargó el programa

GRÁFICO 9:Paso1-Instalación de Maven



Elaborado por: Carlos Medina, Fernando Flores

Fuente: <https://maven.apache.org/docs/3.3.9/release-notes.html>

GRÁFICO 10:Paso2-Instalación de Maven

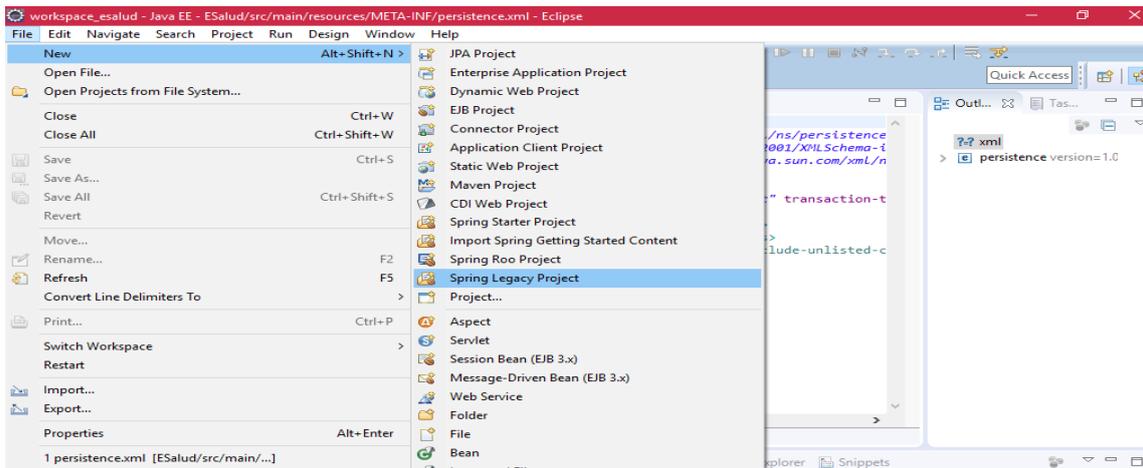


Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Creación de un Proyecto en Eclipse Neón

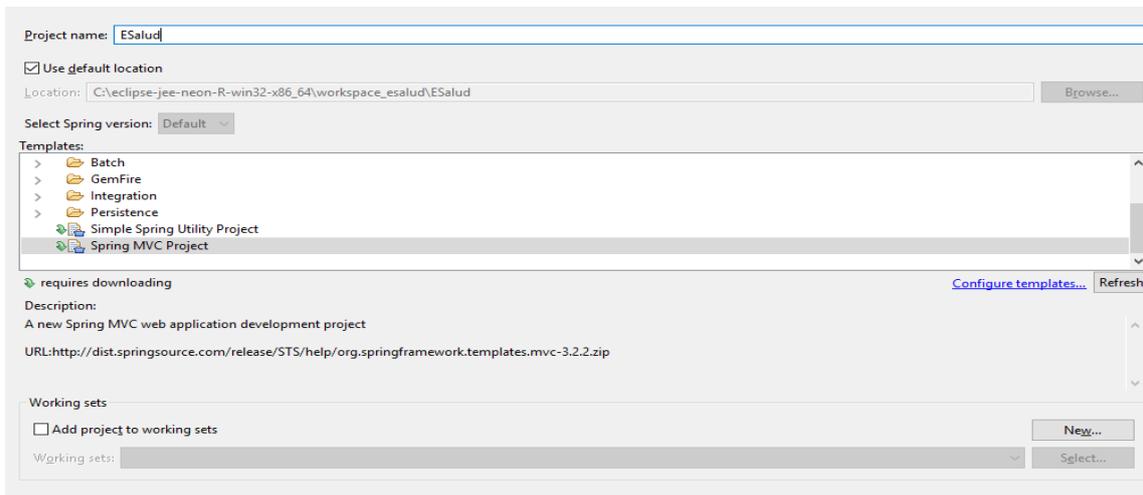
GRÁFICO 11:Paso1-Creación de un Proyecto Spring MVC



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

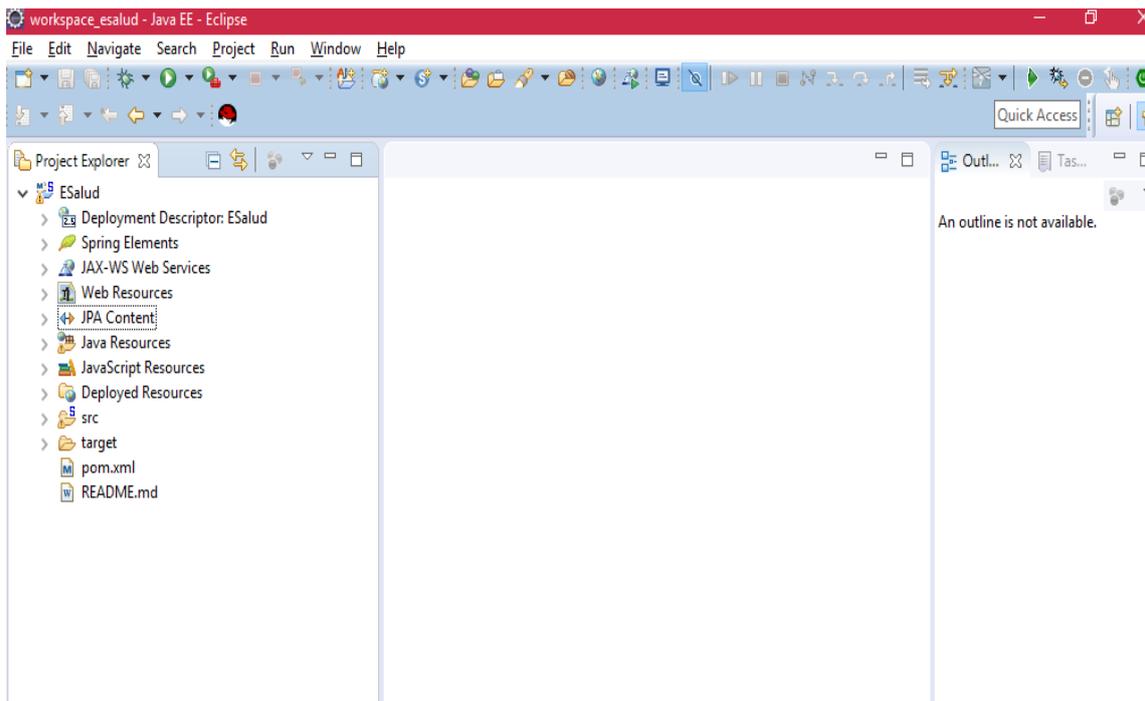
GRÁFICO 12:Paso2-Creación de un Proyecto Spring MVC



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

GRÁFICO 13:Paso3-Creación de un Proyecto Spring MVC



Elaborado por: Carlos Medina, Fernando Flores

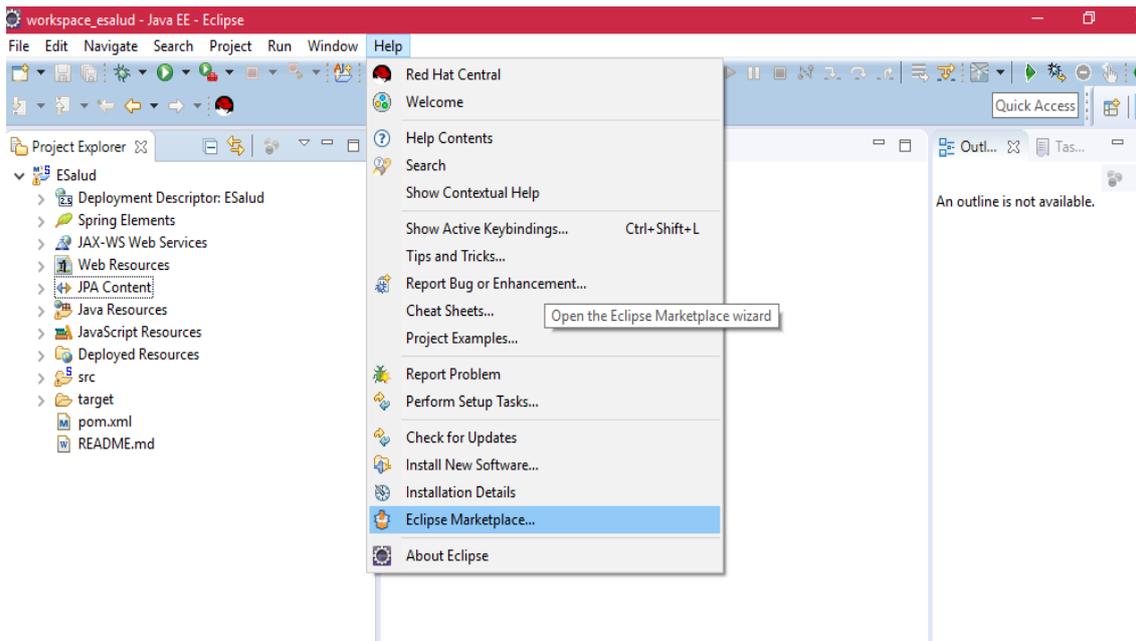
Fuente: Datos de Investigación

Instalación de Spring

Para poder añadir Spring al Proyecto se deben seguir los siguientes pasos:

- 1.- Ejecutar el entorno de desarrollo Eclipse Neón
- 2.- Dar click en el menú Help (Ayuda)
- 3.- Dentro del menú Help, damos click en la opción Eclipse Marketplace
- 4.- Se levantara una ventana en la cual escribimos Spring en el cuadro de texto Find y damos clic en el botón Go.
- 5.- Nos aparecerá opción para descargar el paquete de Spring, damos clic en el botón install.

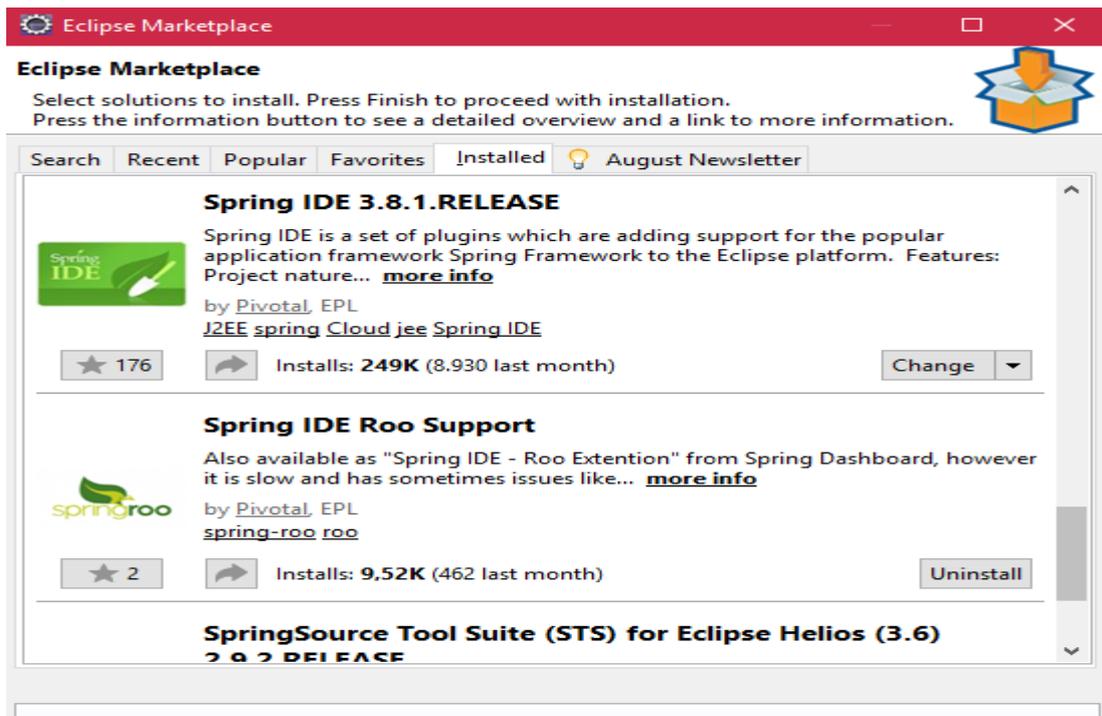
GRÁFICO 14:Paso1-Instalación de Spring



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

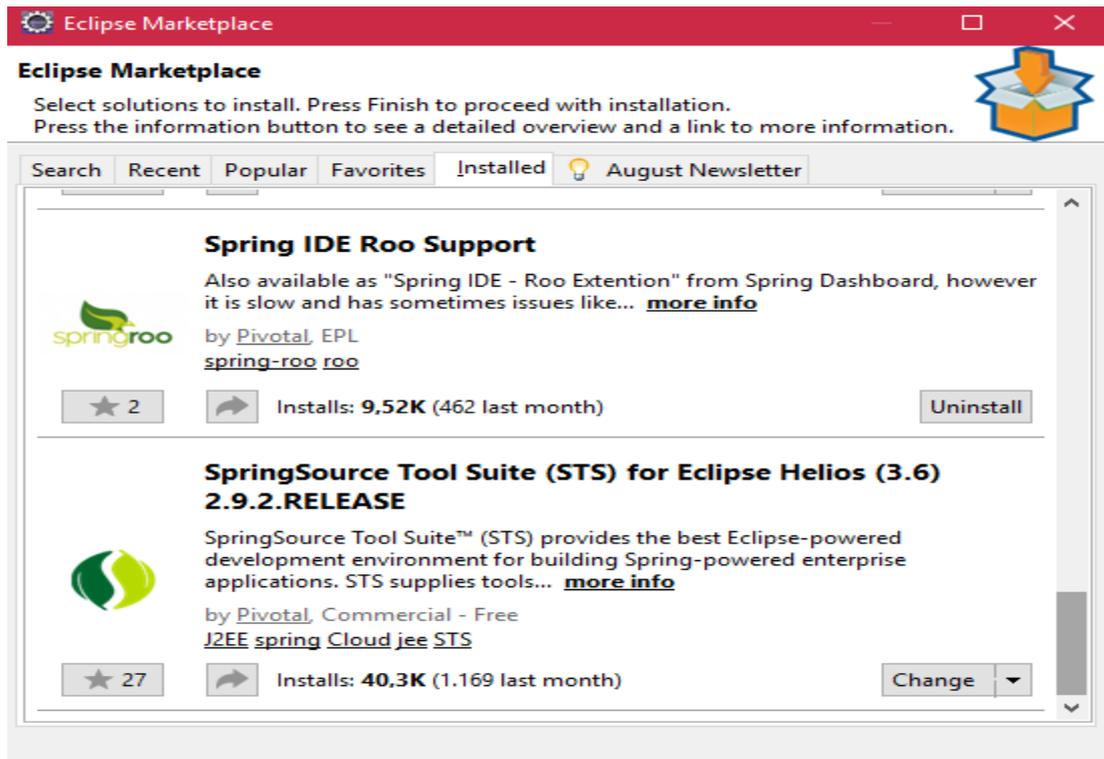
GRÁFICO 15: Paso2-Instalación de Spring



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

GRÁFICO 16: Paso3-Instalación de Spring



Elaborado por: Carlos Medina, Fernando Flores

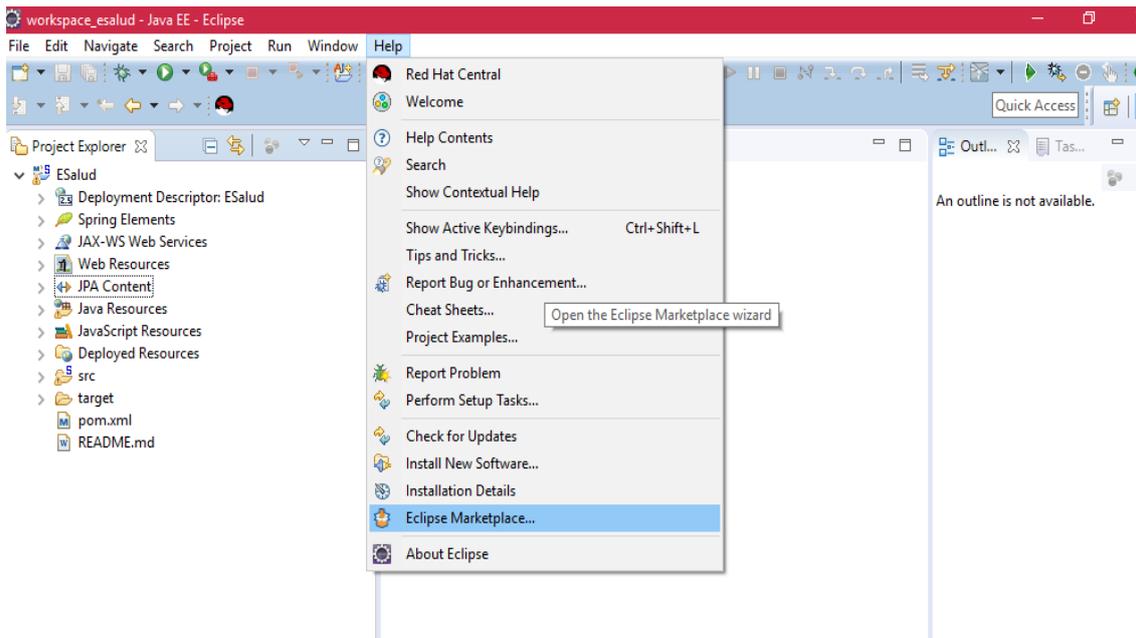
Fuente: Datos de Investigación

Instalación de Jboss Tools 4.4.0.Final

Para poder añadir Jboss Tools al Proyecto se deben seguir los siguientes pasos:

- 1.- Ejecutar el entorno de desarrollo Eclipse Neón
- 2.- Dar clic en el menú Help (Ayuda)
- 3.- Dentro del menú Help, damos clic en la opción Eclipse MarketPlace
- 4.- Se levantara una ventana en la cual escribimos Jboss Tools en el cuadro de texto Find y damos clic en el botón Go.
- 5.- Nos aparecerá opción para descargar el paquete de Jboss Tools, damos clic en el botón Install.

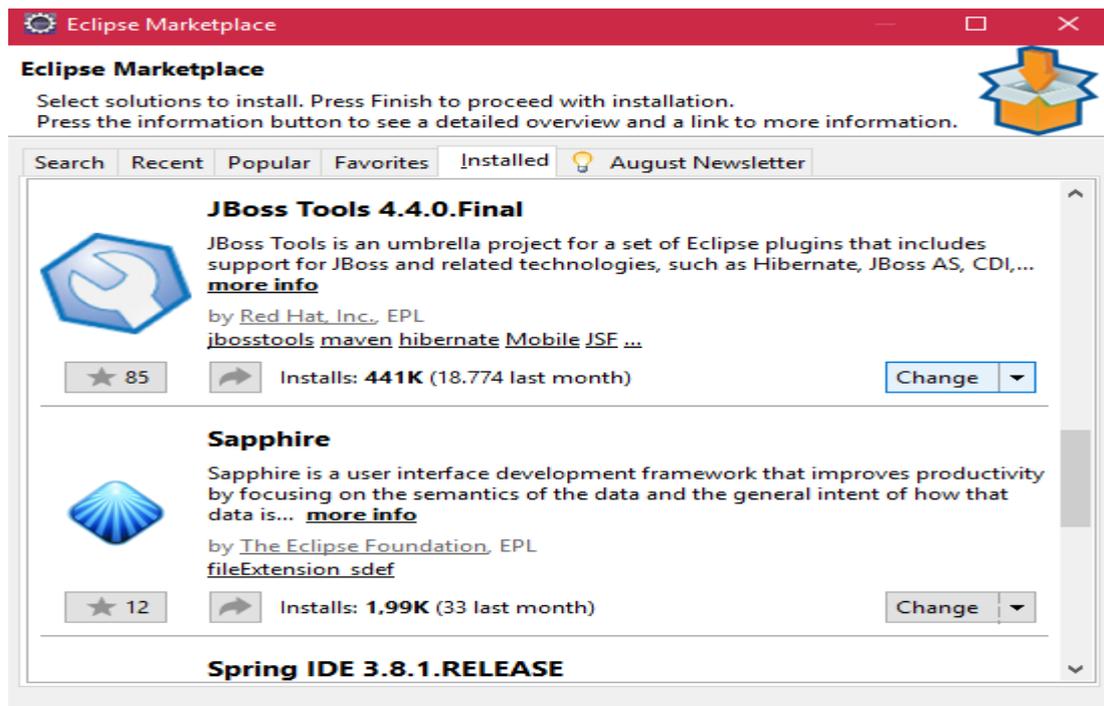
GRÁFICO 17: Paso1- Instalación de Jboss Tools 4.4.0.Final



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

GRÁFICO 18: Paso2- Instalación de Jboss Tools 4.4.0.Final



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Instalación del Servidor Wildfly

Para instalar el servidor Wildfly se siguen los siguientes pasos:

1.-Nos dirigimos a la página oficial para descarga el software:

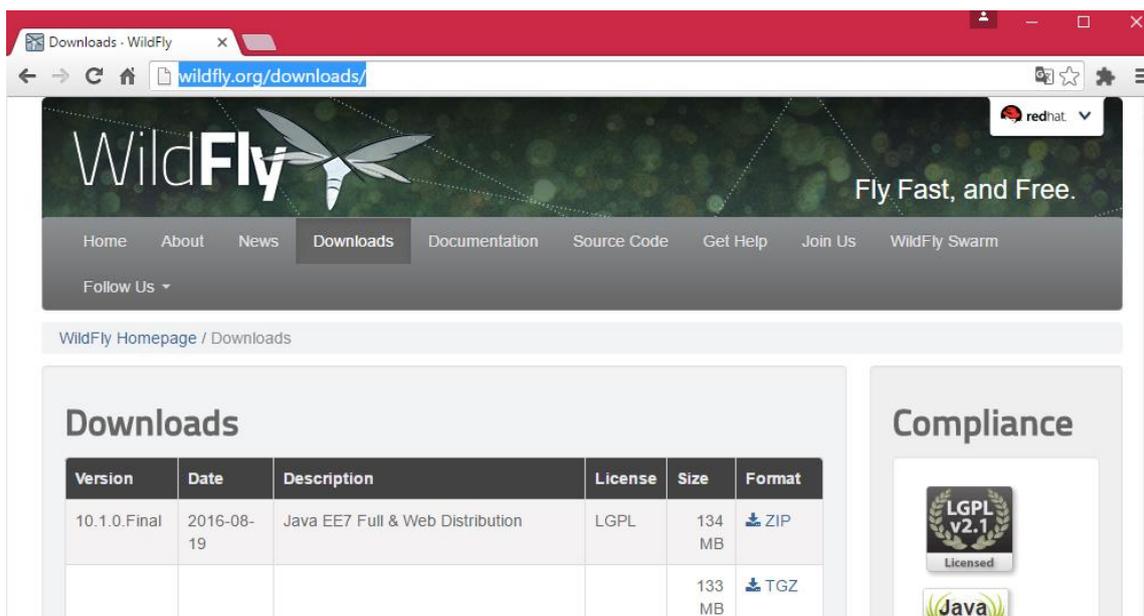
<http://wildfly.org/downloads/>

2.-Escogemos la versión para nuestro sistema operativo y seleccionamos descargar.

3. Nos ubicamos en la ruta donde se descargó el programa

4.- Configuramos la ruta donde queremos que se aloje la carpeta.

GRÁFICO 19: Instalación del Servidor Wildfly



Elaborado por: Carlos Medina, Fernando Flores

Fuente: <http://wildfly.org/downloads/>

Configurar la Variable de entorno Java Home y Path

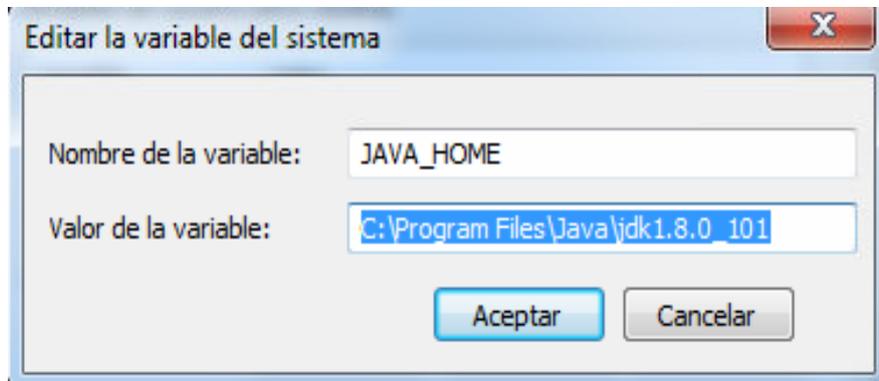
Paso1.- Configurar la Variable Java Home

JAVA_HOME, es una variable de entorno del sistema que informa al sistema operativo sobre la ruta donde se encuentra instalado Java.

Seguiremos la siguiente secuencia de pasos para configurar esta variable:

Abrimos el explorador de Windows o pulsamos sobre “Mi Pc”. Pulsamos sobre Equipo y con botón derecho del ratón o buscando el icono -> Propiedades -> Configuración avanzada / Cambiar configuración -> Opciones avanzadas -> Variables de entorno -> Nueva (Variables del sistema).

GRÁFICO 20: Paso1-Configurar Variable Java Home



Elaborado por: Carlos Medina, Fernando Flores

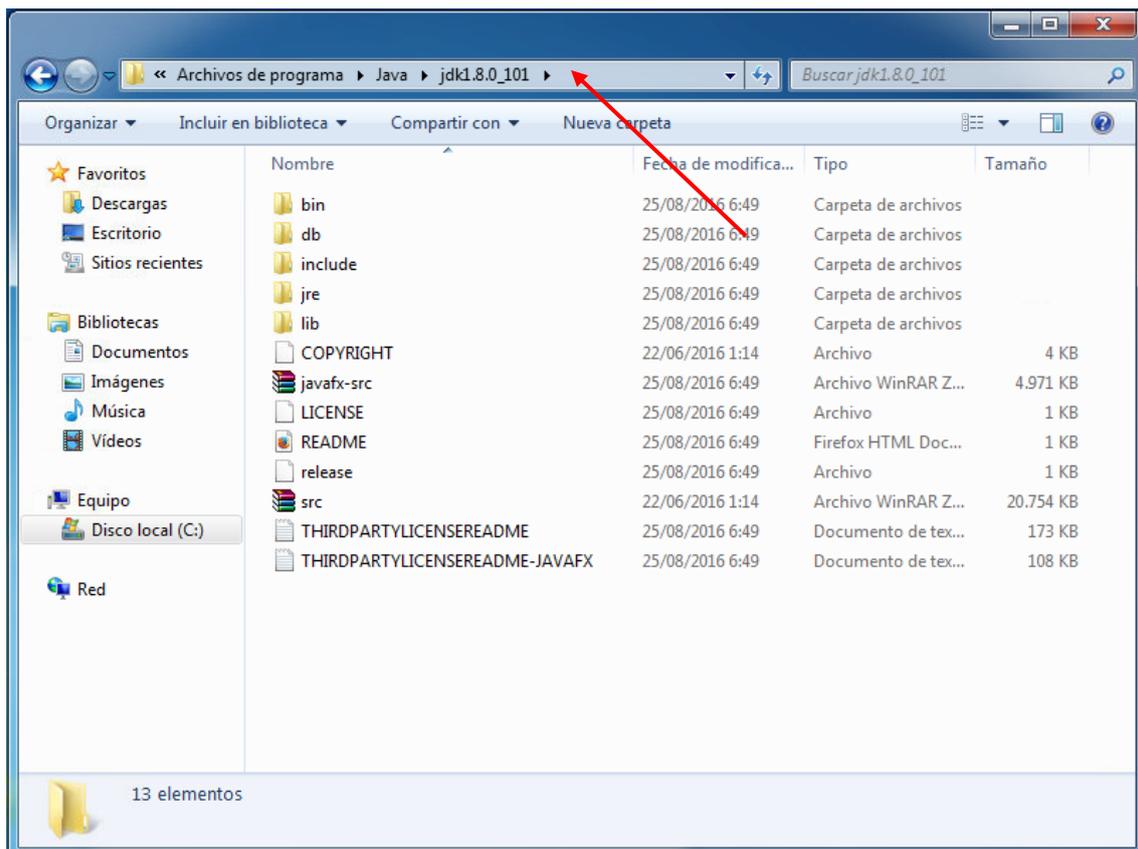
Fuente:http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=389:configurar-java-en-windows-variables-de-entorno-javahome-y-path-cu00610b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188

Escribiremos en las cajas de texto que se muestran lo siguiente:

Nombre de variable: JAVA_HOME

Valor de variable: escribiremos aquí la ruta en que se haya instalado Java. Puedes consultarla en el propio explorador de Windows buscando la carpeta en que se ha instalado Java, que normalmente será del tipo C:\Program Files\Java\jdk1.8.0_72 ó C:\Program Files (x86)\Java\jdk1.8.0_72 o similar

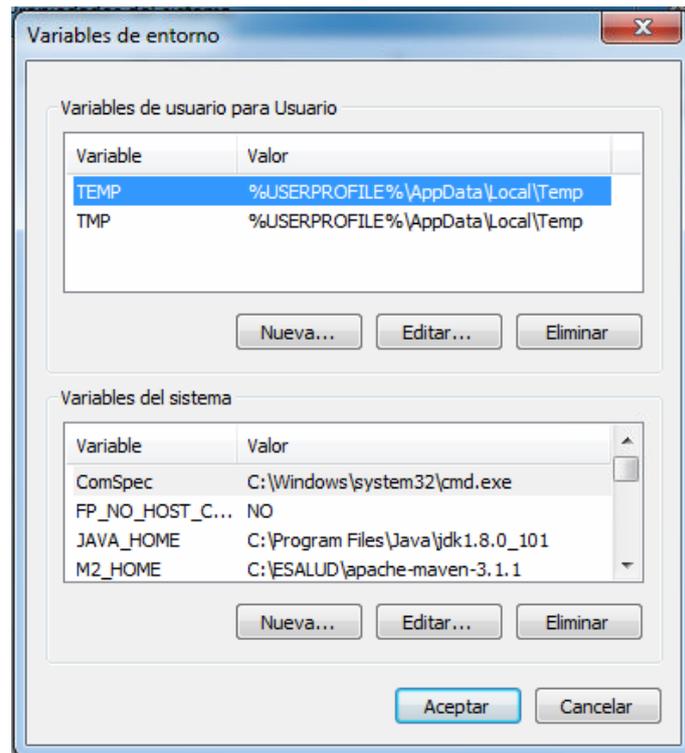
GRÁFICO 21: Paso2-Configurar Variable Java Home



Elaborado por: Carlos Medina, Fernando Flores

Fuente: http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=389:configurar-java-en-windows-variables-de-entorno-javahome-y-path-cu00610b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188

GRÁFICO 22: Paso3-Configurar Variable Java Home



Elaborado por: Carlos Medina, Fernando Flores

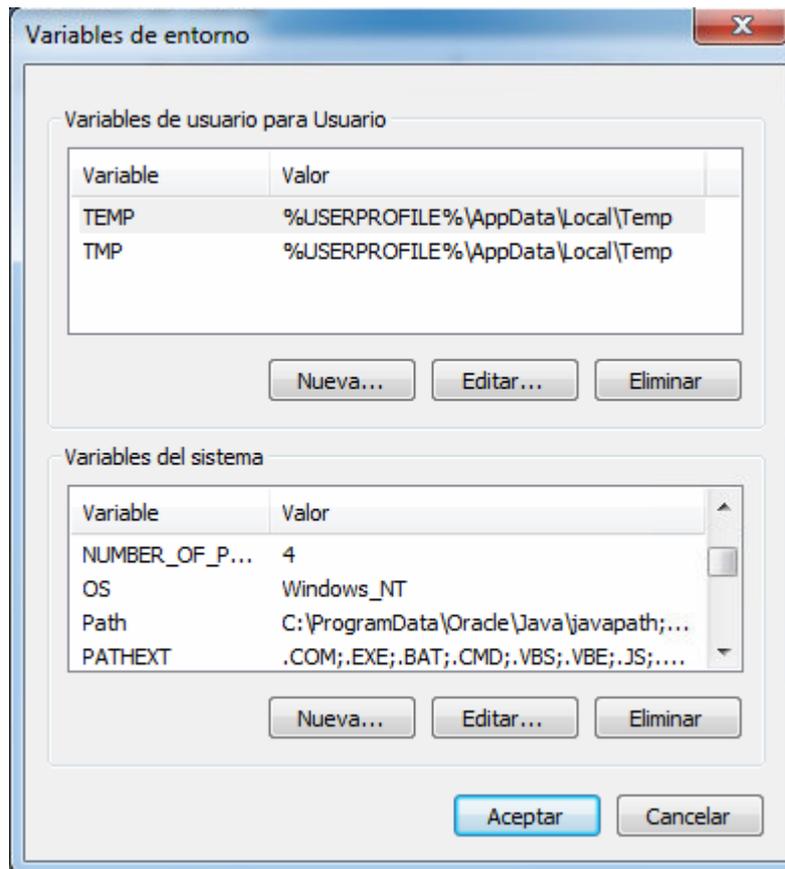
Fuente:http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=389:configurar-java-en-windows-variables-de-entorno-javahome-y-path-cu00610b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188

Paso 2.- Configurar la Variable Path

PATH es una variable de entorno del sistema que informa al sistema operativo sobre la ruta de distintos directorios esenciales para el funcionamiento del ordenador.

Abrimos el explorador de Windows o pulsamos sobre "Mi Pc". Pulsamos sobre Equipo -> Propiedades y con botón derecho del ratón o buscando el icono -> Configuración avanzada / Cambiar configuración -> Opciones avanzadas -> Variables de entorno -> buscamos en Variables del sistema la variable Path, pulsamos para seleccionarla y hacemos click en el botón Editar. Luego al final del contenido que ya exista, añadiremos un punto y coma y el texto %JAVA_HOME%\bin. No deben quedar espacios intermedios.

GRÁFICO 23: Configurar Variable Path



Elaborado por : Carlos Medina, Fernando Flores

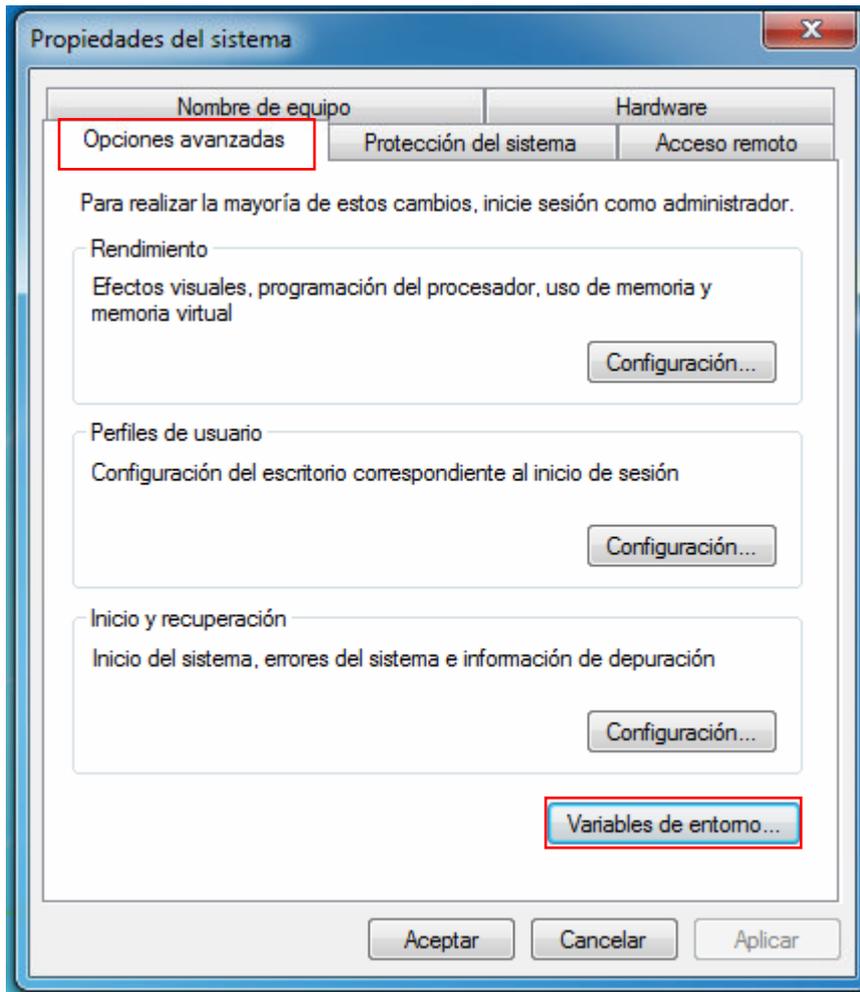
Fuente:http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=389:configurar-java-en-windows-variables-de-entorno-javahome-y-path-cu00610b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188

Configurar Maven en Windows

Para poder utilizar maven necesitaremos tener configuradas dos variables de entorno: MAVEN_HOME y PATH

Inicio > Mi PC > (click derecho) > Propiedades > (Pestaña) Opciones avanzadas > (Botón) variables de entorno

GRÁFICO 24: Paso1- Configurar Maven en Windows

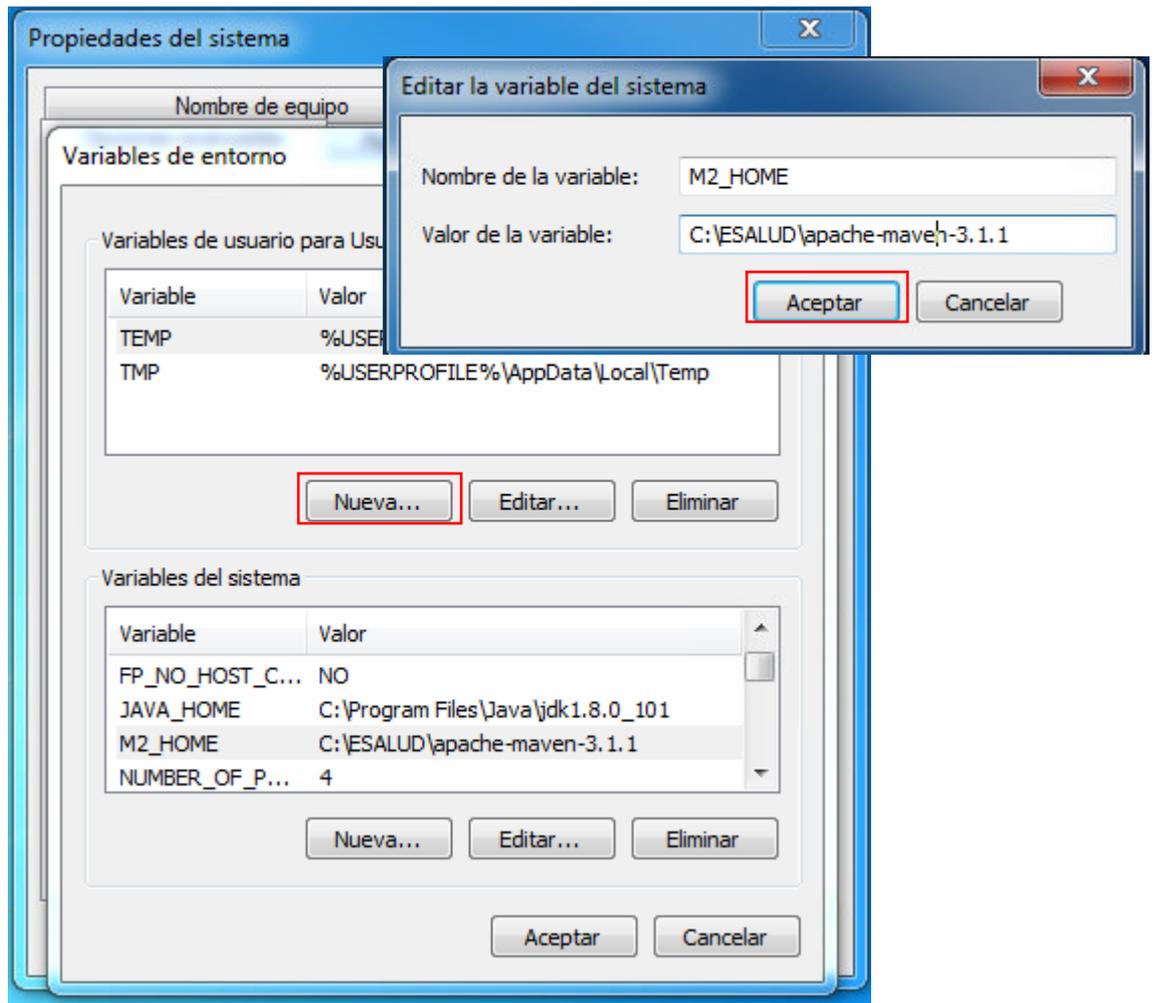


Elaborado por: Carlos Medina, Fernando Flores

Fuente: <http://tutorialesparanovatos.blogspot.com/2013/04/en-este-tutorial-veras-como-instalar.html>

Para M2_HOME ingresaremos la ruta donde instalamos nuestro maven, es decir
M2_HOME = p:\apache-maven-3.0.5

GRÁFICO 25: Paso2- Configurar Maven en Windows



Elaborado por: Carlos Medina, Fernando Flores **Fuente:**

<http://tutorialesparanovatos.blogspot.com/2013/04/en-este-tutorial-veras-como-instalar.html>

Luego editaremos nuestra (o crearemos de ser necesario) nuestra variable PAHT

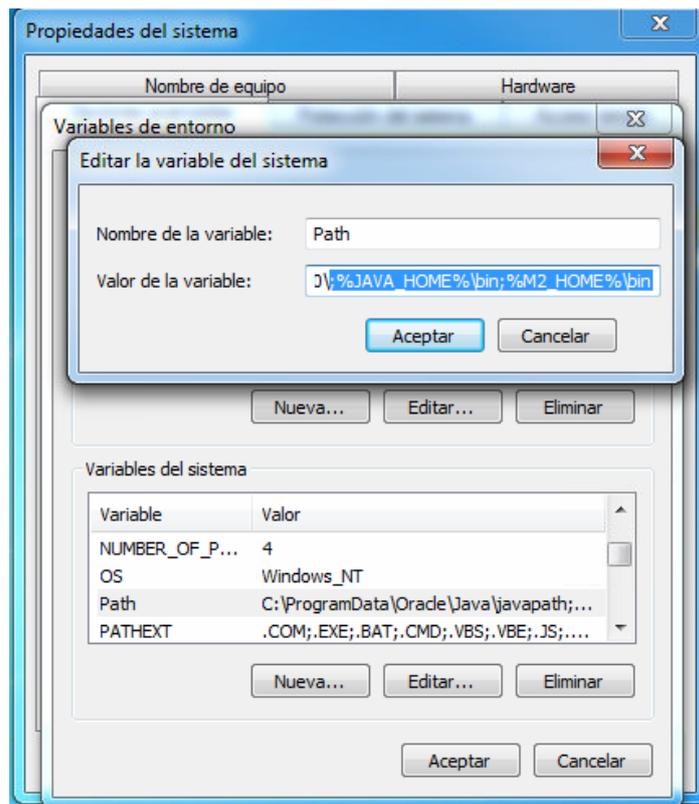
Si la variable no existe la creamos igual que lo hicimos con la anterior, y le asignamos el siguiente valor:

PAHT = %M2_HOME%\bin

Si ya tienes creada la variable PAHT, simplemente le concatenas al final la ruta anteponiendo ';', osea que tendremos algo parecido a esto:

PAHT = variable1;variable2;%M2_HOME%\bin

GRÁFICO 26: Paso3- Configurar Maven en Windows



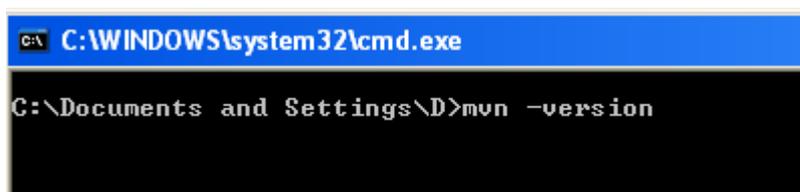
Elaborado por : Carlos Medina, Fernando Flores

Fuente: <http://tutorialesparanovatos.blogspot.com/2013/04/en-este-tutorial-veras-como-instalar.html>

Comprobamos la instalación

En abrimos la consola en Inicio > Ejecutar >cmd e ingresamos el comando mvn -version

GRÁFICO 27: Paso4- Configurar Maven en Windows



Elaborado por : Carlos Medina, Fernando Flores

Fuente: <http://tutorialesparanovatos.blogspot.com/2013/04/en-este-tutorial-veras-como-instalar.html>

Instalación de PostgreSQL

Para instalar PostgreSQL se siguen los siguientes pasos:

1.-Nos dirigimos a la página para descarga el software:

<http://www.filecroco.com/download-postgresql-64>

2.-Escogemos la versión para nuestro sistema operativo y seleccionamos descargar.

3. Nos ubicamos en la ruta donde se descargó el programa y le damos ejecutar

4.- Configuramos la ruta donde queremos que se instale el programa.

GRÁFICO 28: Paso1- Instalación de PostgreSQL



PostgreSQL 9.5.3 (64-bit) free download **61 MB**

Date added: May, 17th 2016 - (Freeware)
★★★★★ (5/5 - 1 votes)

Features **Change Log** Me gusta 1 Tweet G+ 0 Share

Current version: 9.5.3
File name: postgresql-9.5.3-1-windows-x64.exe
MD5 checksum: EDFEEB34CB7FBEA317AE209BD42E0CB0
File size: 61MB
License: Freeware
Author / Homepage: PostgreSQL Global Development / www.postgresql.org

PostgreSQL benefits from 15 years of active development that have led to unlimited database size, unlimited rows per table and unlimited indexes per table. As Open Source object relational database system, it represents a pinnacle in cross-platform integration, ACID compliance and foreign keys, triggers, procedures or joins.

With native programming interfaces and a complete documentation on everything you need to do, this software remains highly customizable in terms of the function library. Additionally, support for sub-queries, transaction isolation levels, multiple schema per database, and the integration of extensions and of the Generalized Search Tree add further flexibility the large array of functions.

Elaborado por: Carlos Medina, Fernando Flores

Fuente: <http://www.filecroco.com/download-postgresql-64>

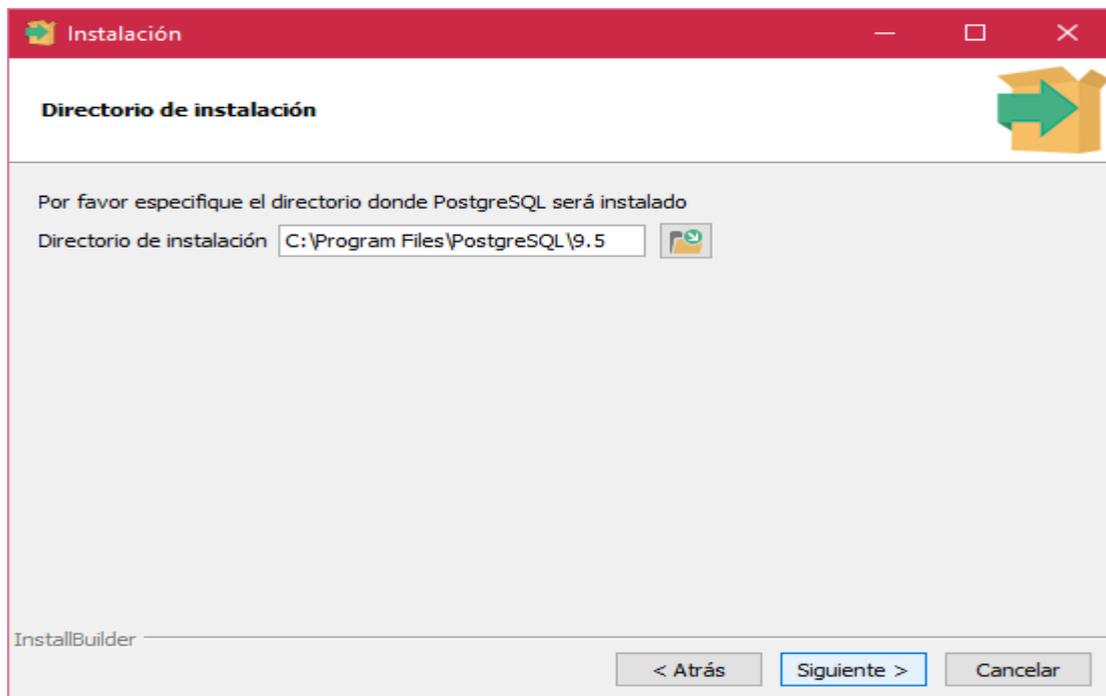
GRÁFICO 29: Paso2- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

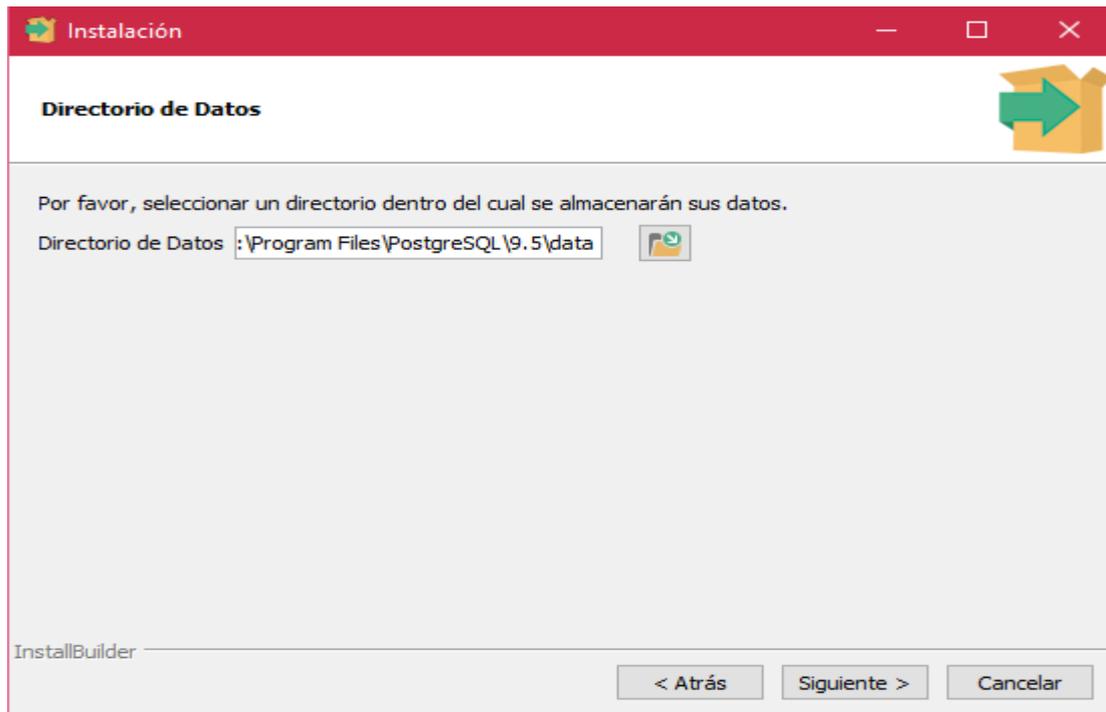
GRÁFICO 30: Paso3- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

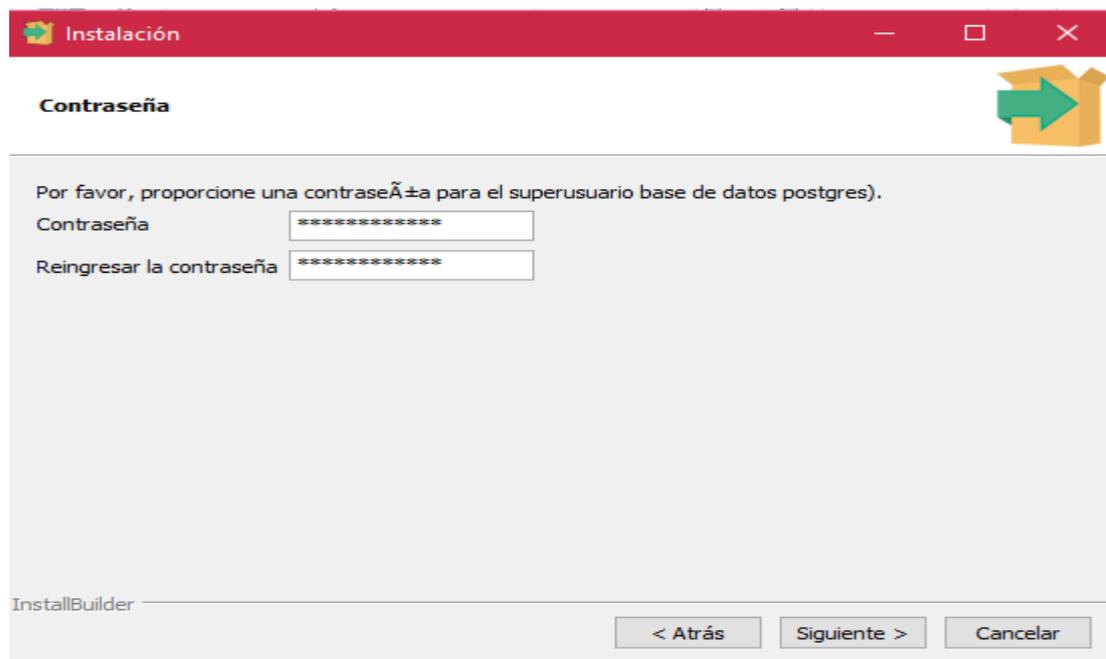
GRÁFICO 31: Paso4- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

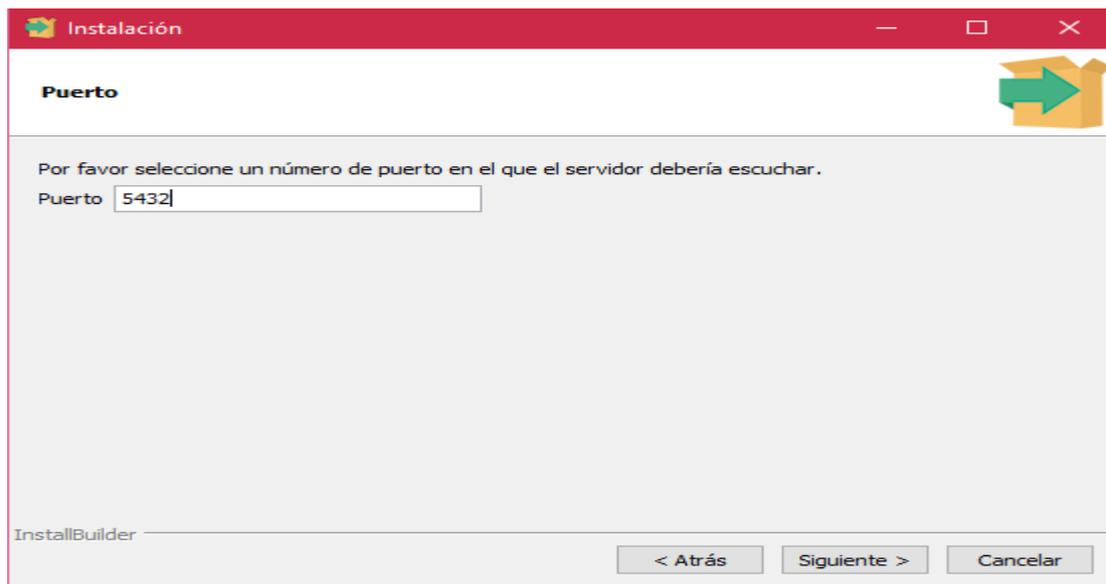
GRÁFICO 32: Paso5- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

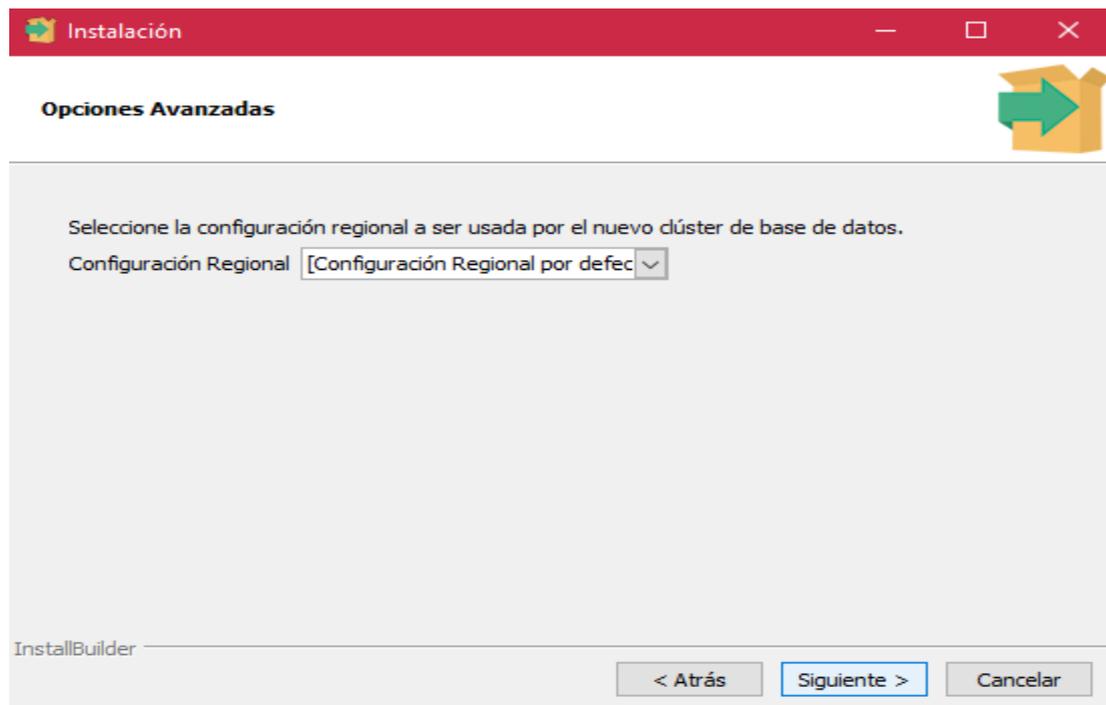
GRÁFICO 33: Paso6- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

GRÁFICO 34: Paso7- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores

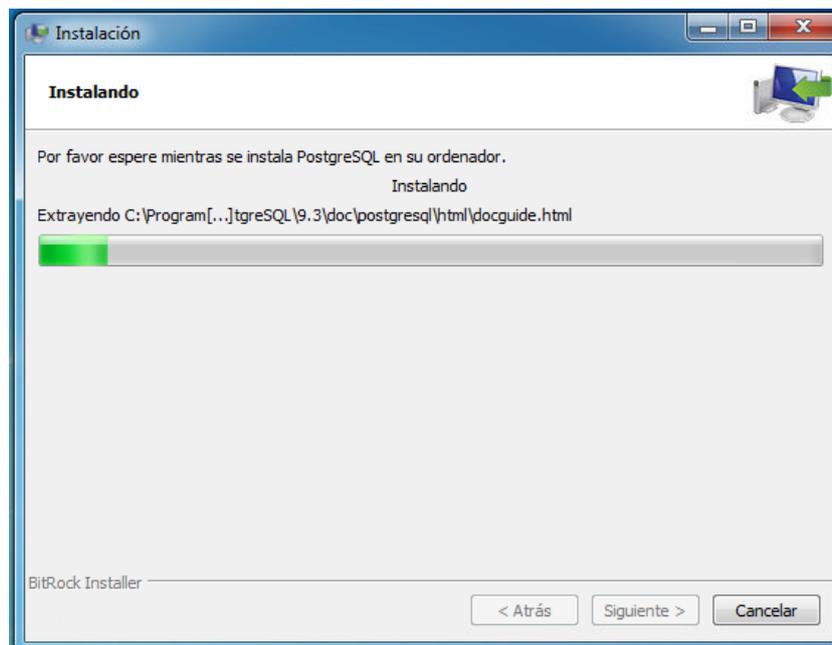
Fuente: Datos de Investigación

GRÁFICO 35: Paso8- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores
Fuente: Datos de Investigación

GRÁFICO 36: Paso9- Instalación de PostgreSQL



Elaborado por: Carlos Medina, Fernando Flores
Fuente: Datos de Investigación

GRÁFICO 37: Paso10- Instalación de PostgreSQL

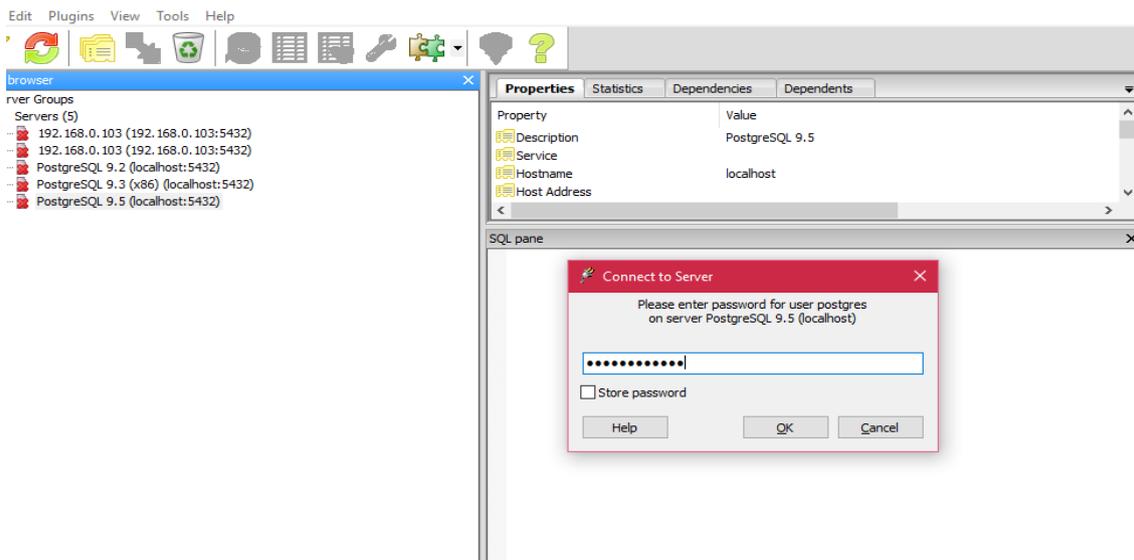


Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Conectándose a la Base de Datos

GRÁFICO 38: Conexión a la Base de Datos

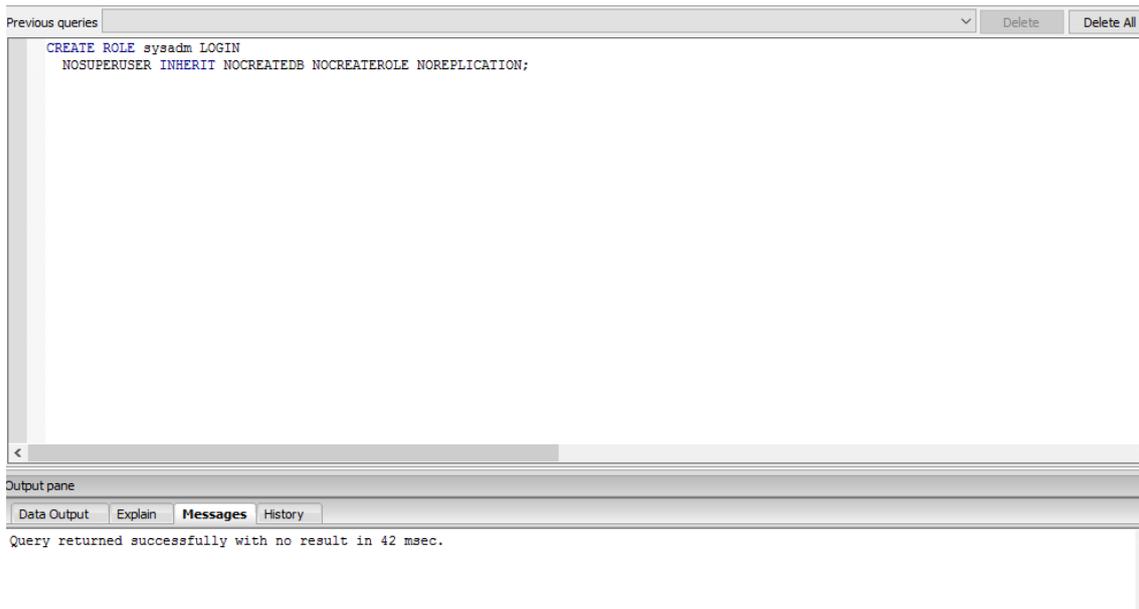


Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Creando un nuevo rol en la Base de Datos

GRÁFICO 39: Creación de Nuevo Rol

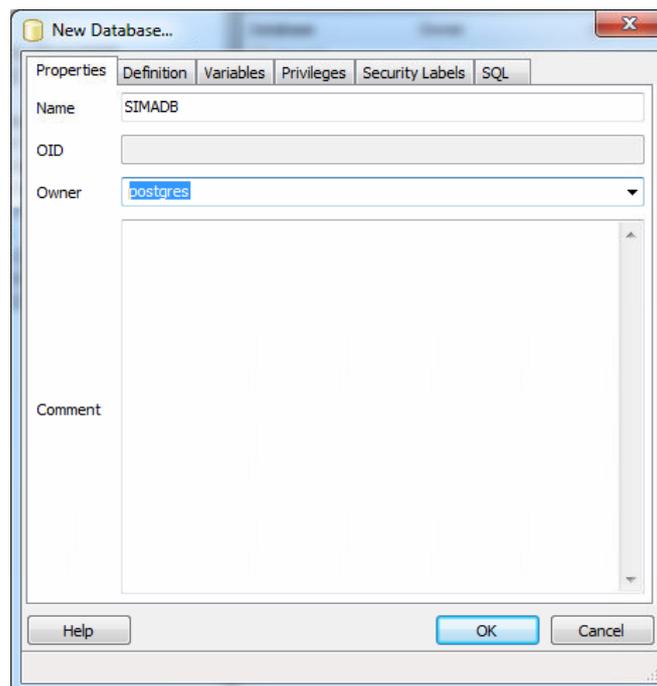


Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Creando la Base de Datos SIMADB

GRÁFICO 40: Creación de la Base de Datos

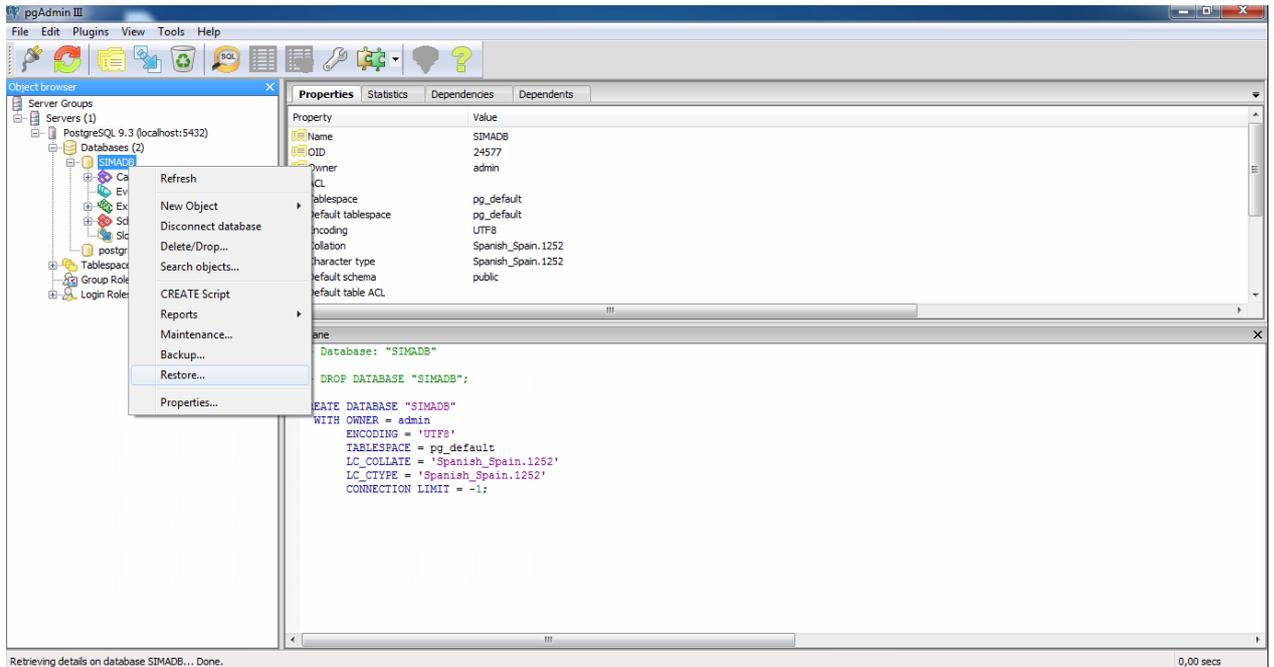


Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Restaurando la Base de Datos

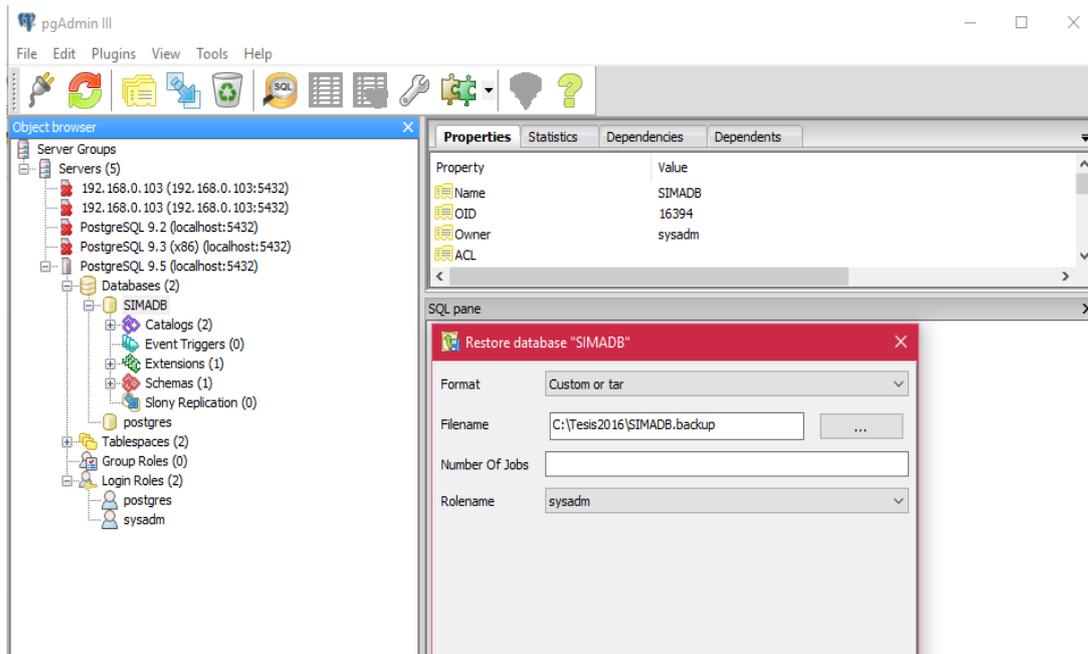
GRÁFICO 41: Paso1- Importar Base de Datos



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

GRÁFICO 42: Paso2- Importar Base de Datos



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

Glosario de Tablas de BD

ffacantones

En esta tabla se encuentran los registros de los cantones ingresados en el sistema.

ffadiagnosticoclas

En esta tabla se encuentran los registros de los tipos de diagnóstico en el sistema.

ffadistrito

En esta tabla se encuentran las descripciones de los distritos del MSP.

fflocalidad

En esta tabla se encuentra registrado las descripciones de los lugares de una ciudad.

Ffavivienda_ubicacion

En esta tabla se encuentran los datos que se ingresó en el formulario de ubicación de la vivienda.

Ffavivienda_intervencion

En esta tabla se encuentra los datos que se ingresó en el formulario de intervenciones familiares.

Ffavivienda_mortalidad

En esta tabla se encuentran los datos que se ingresó en el formulario de mortalidad familiar de las personas.

Ffavivienda_diagnostico

En esta tabla se encuentran los datos que se ingresó en el formulario de diagnóstico familiar.

ffaviviendaagtrat

En esta tabla se encuentran los tipos de tratamiento de agua que le da la familia.

ffaviviendaalumbrado

En esta tabla se encuentran quien le brinda servicio de alumbrado público a la familia.

ffaviviendabasura

En esta tabla se encuentran la descripción del tipo de empresa que ofrece la recolección de la basura a la familia.

ffaviviendaentrevista

En esta tabla se encuentra los tipos de entrevistas que se realizan a la familia.

ffaviviendacocina

Esta tabla se encuentra los tipos de cocinas que la familia tiene en su hogar.

ffaviviendapiso

En esta tabla se encuentra los tipos de piso con que puede contar la familia en su hogar.

ffaviviendariesgossanitario

En esta tabla se encuentran los tipos de riesgos sanitarios que puede tener la familia.

Ffaviviendash

Esta tabla se encuentra los tipos de servicios higiénicos que puede tener la familia.

ffaviviendatecho

En esta tabla se encuentra los tipos de techos que puede tener la familia.

ffaviviendatiempo

En esta tabla se encuentran la cantidad de tiempo que le toma a la familia de ir al trabajo.

ffaviviendatransporte

En esta tabla se encuentran los tipos de transportes que puede tomar la familia para llegar al hogar.

ffazona

En esta tabla se encuentra la descripción de las zonas configuradas por el MSP.

hcflocalizacion

En esta tabla se encuentra la descripción de la Ubicación.

Código Fuente del Sistema de Ficha Familiar

Vistas del Sistema Ficha Familiar

Vista Datos del Paciente

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

    <ui:composition template="/WEB-INF/template.xhtml">
        <ui:define name="title">FICHA FAMILIAR</ui:define>
    <ui:define name="content">
        <h:form id="form1">
            <p:growl id="growl" showDetail="true" showSummary="true"
                sticky="true" />
            <div align="left">

                <h:panelGrid columns="4" cellpadding="5">
                    <p:inputText id="datosConsulta"
size="30"
value="#{pacienteMB.cedula}" />
                    <p:watermark for="datosConsulta"
value="Buscar por
identificacion" />
                    <p:commandButton
value="Consultar" id="btnConsultar"
actionListener="#{pacienteMB.buscarPaciente}" update="form1" />
                    <p:commandButton value="Limpiar"
id="btnLimpiar"
actionListener="#{pacienteMB.limpiar}" update="form1" />
                </h:panelGrid>

                <h:panelGrid columns="1" cellpadding="5" width="100%">
                    <p:panel width="80%" header="Datos de
Ubicacion de La Vivienda">
                        <h:panelGrid columns="5"
cellpadding="5">
                            <p:column width="50">
                                <p:outputLabel
value="Identificacion" />
                            </p:column>
                            <p:column width="50">
                                <p:inputText size="50"
value="#{pacienteMB.ident}"
disabled="#{pacienteMB.habpaciente}"/>
                        </h:panelGrid>
                    </p:panel>
                </h:panelGrid>
            </div>
        </h:form>
    </ui:define>
</ui:composition>
```

```

        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel
value="Primer Nombre" />

        </p:column>
        <p:column width="50">
            <p:inputText size="50"
value="#{pacienteMB.pnomb}"
disabled="#{pacienteMB.habpaciente}"/>
        </p:column>

        <p:column width="50">
            <p:outputLabel
value="Segundo Nombre" />

        </p:column>
        <p:column width="50">
            <p:inputText size="50"
value="#{pacienteMB.snomb}"
disabled="#{pacienteMB.habpaciente}"/>
        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel
value="Apellido Paterno" />

        </p:column>
        <p:column width="50">
            <p:inputText size="50"
value="#{pacienteMB.papell}"
disabled="#{pacienteMB.habpaciente}"/>
        </p:column>

        <p:column width="50">
            <p:outputLabel
value="Apellido Materno" />

        </p:column>
        <p:column width="50">
            <p:inputText size="50"
value="#{pacienteMB.sapell}"
disabled="#{pacienteMB.habpaciente}"/>
        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel
for="establecimiento" value="Identificador del Establecimiento" />

```

```

        </p:column>
        <p:column width="50">
            <p:selectOneMenu id="establecimiento"
value="#{pacienteMB.establecimiento}" style="width:350px"
                effect="fold"
            editable="true" disabled="#{pacienteMB.habpaciente}">
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
                <f:selectItems var="Loc"
value="#{pacienteMB.lstLocalizacionEstab}"
                    itemLabel="#{Loc.nombre}"
                    itemValue="#{Loc.nombre}"
                />
            </p:selectOneMenu>
        </p:column>
        <p:column width="50">
            <p:outputLabel
for="provincia" value="Provincia" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu
id="provincia" value="#{pacienteMB.provinc}" style="width:350px"
                effect="fold"
            editable="true" disabled="#{pacienteMB.habpaciente}">
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
                <f:selectItems var="Local"
value="#{pacienteMB.lstLocalizacion}"
                    itemLabel="#{local.ubicacion}"
                    itemValue="#{local.ubicacion}"/>
            </p:selectOneMenu>
        </p:column>
        <p:spacer width="50" height="20"
/>
        <p:column width="50">
            <p:outputLabel
for="canton" value="Canton" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu
id="canton" value="#{pacienteMB.canton}" style="width:350px"
                effect="fold"
            editable="true" disabled="#{pacienteMB.habpaciente}">
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
                <f:selectItems var="cantones"
value="#{pacienteMB.lstLocalizacionCanton}"
                    itemLabel="#{cantones.descripcion}"
            </p:selectOneMenu>
        </p:column>

```

```

itemValue="#{cantones.descripcion}"/>
</p:selectOneMenu>
</p:column>
<p:column width="50">
  <p:outputLabel
for="parroquia" value="Parroquia" />
</p:column>
<p:column width="50">
  <p:selectOneMenu
id="parroquia" value="#{pacienteMB.parroquia}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
    <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
    <f:selectItems var="par"
value="#{pacienteMB.Lstparroquia}"
itemLabel="#{par.descripcion}"
itemValue="#{par.descripcion}" />
  </p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
  <p:outputLabel
for="distrito" value="Distrito" />
</p:column>
<p:column width="50">
  <p:selectOneMenu
id="distrito" value="#{pacienteMB.distrito}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
    <f:selectItem itemLabel="Seleccione Uno"
itemValue="" />
    <f:selectItems var="dis"
value="#{pacienteMB.Lstdistrito}"
itemLabel="#{dis.descripcion}"
itemValue="#{dis.descripcion}" />
  </p:selectOneMenu>
</p:column>
<p:column width="50">
  <p:outputLabel for="zona"
value="Zona" />
</p:column>
<p:column width="50">
  <p:selectOneMenu
id="zona" value="#{pacienteMB.zona}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">

```

```

itemValue="" />
value="#{pacienteMB.lstzona}"
itemLabel="#{z.numero}"
itemValue="#{z.numero}" />
/>

value="Localidad" />

id="Ldad" value="#{pacienteMB.localidad}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}"
<f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
value="#{pacienteMB.lstlocal}"
itemLabel="#{ld.descripcion}"
itemValue="#{ld.descripcion}" />

value="Sector" />

value="#{pacienteMB.sector}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}"
<f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
value="#{pacienteMB.lstsector}"
itemLabel="#{sc.numero}"
itemValue="#{sc.numero}" />
/>

</h:panelGrid>
</p:panel>

```

```

Vivienda">
    <p:panel width="80%" header="Direccion
        <h:panelGrid columns="5"
            cellpadding="5">
                <p:column width="50">
                    <p:outputLabel for="mnz"
                        value="Manzana" />
                </p:column>
                <p:column width="50">
                    <p:selectOneMenu id="mnz"
                        value="#{pacienteMB.manzana}" style="width:350px"
                        effect="fold"
                        editable="true" disabled="#{pacienteMB.habpaciente}">
                        <f:selectItem itemLabel="Seleccione
                            Uno" itemValue="" />
                        <f:selectItems var="mz"
                            value="#{pacienteMB.lstmanzana}"
                            itemLabel="#{mz.descripcion}"
                            itemValue="#{mz.descripcion}" />
                    </p:selectOneMenu>
                </p:column>
                <p:spacer width="50" height="20"
                    />
                <p:column width="50">
                    <p:outputLabel for="dv"
                        value="Division" />
                </p:column>
                <p:column width="50">
                    <p:selectOneMenu id="dv"
                        value="#{pacienteMB.division}" style="width:350px"
                        effect="fold"
                        editable="true" disabled="#{pacienteMB.habpaciente}">
                        <f:selectItem itemLabel="Seleccione
                            Uno" itemValue="" />
                        <f:selectItems var="dsn"
                            value="#{pacienteMB.lstdivision}"
                            itemLabel="#{dsn.descripcion}"
                            itemValue="#{dsn.descripcion}"/>
                    </p:selectOneMenu>
                </p:column>
                <p:column width="50">
                    <p:outputLabel for="edif"
                        value="Edificio" />
                </p:column>
                <p:column width="50">
                    <p:selectOneMenu
                        id="edif" value="#{pacienteMB.edificio}" style="width:350px"
                        effect="fold"
                        editable="true" disabled="#{pacienteMB.habpaciente}">
                        <f:selectItem itemLabel="Seleccione
                            Uno" itemValue="" />
                    </p:selectOneMenu>
                </p:column>
            </h:panelGrid>
        </p:panel>
    </div>

```

```

value="#{pacienteMB.Lstedificio}"
itemLabel="#{edf.descripcion}"
itemValue="#{edf.descripcion}"/>
/>

value="Numero de Vivienda" />

id="numv" value="#{pacienteMB.numvivienda}" style="width:350px"
editable="true" disabled="#{pacienteMB.habpaciente}"
Uno" itemValue="" />

value="#{pacienteMB.Lstvivnumero}"
itemLabel="#{vnm.descripcion}"
itemValue="#{vnm.descripcion}"/>

value="Numero de Hogar" />

value="#{pacienteMB.numhogar}" style="width:350px"
editable="true" disabled="#{pacienteMB.habpaciente}"
Uno" itemValue="" />

value="#{pacienteMB.Lstvivhogar}"
itemLabel="#{nhg.descripcion}"
itemValue="#{nhg.descripcion}"/>
/>

value="Calle Principal" />

```

```

<f:selectItems var="edf"

    </p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"

<p:column width="50">
    <p:outputLabel for="numv"

</p:column>
<p:column width="50">
    <p:selectOneMenu
        effect="fold"
    <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
    <f:selectItems var="vnm"

    </p:selectOneMenu>
</p:column>
<p:column width="50">
    <p:outputLabel for="hg"

</p:column>
<p:column width="50">
    <p:selectOneMenu id="hg"
        effect="fold"
    <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
    <f:selectItems var="nhg"

    </p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"

<p:column width="50">
    <p:outputLabel

</p:column>
<p:column width="50">

```

```

value="#{pacienteMB.pcalle}"
disabled="#{pacienteMB.habpaciente}"/>
</p:column>
<p:column width="50">
  <p:outputLabel
value="Calle Secundaria" />
</p:column>
<p:column width="50">
  <p:inputText size="50"
value="#{pacienteMB.scalle}"
disabled="#{pacienteMB.habpaciente}"/>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
  <p:outputLabel
value="Numero de Casa" />
</p:column>
<p:column width="50">
  <p:inputText size="50"
value="#{pacienteMB.numcasa}"
disabled="#{pacienteMB.habpaciente}"/>
</p:column>
<p:column width="50">
  <p:outputLabel
value="Conjunto Habitacional" />
</p:column>
<p:column width="50">
  <p:inputText size="50"
value="#{pacienteMB.conjhabitac}"
disabled="#{pacienteMB.habpaciente}"/>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
  <p:outputLabel
value="Numero de Lote" />
</p:column>
<p:column width="50">
  <p:inputText size="50"
value="#{pacienteMB.numLote}"
disabled="#{pacienteMB.habpaciente}"/>
</p:column>
<p:column width="50">

```

```

value="Numero de Departamento" />
    <p:outputLabel
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
            value="#{pacienteMB.numdepart}"
            disabled="#{pacienteMB.habpaciente}"/>
    </p:column>
    <p:spacer width="50" height="20" />
    <p:column width="50">
        <p:outputLabel
            value="Numero de Piso" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
            value="#{pacienteMB.numpiso}"
            disabled="#{pacienteMB.habpaciente}"/>
    </p:column>
    <p:column width="50">
        <p:outputLabel
            value="Telefono" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
            value="#{pacienteMB.telefono}"
            disabled="#{pacienteMB.habpaciente}"/>
    </p:column>
    <p:spacer width="50" height="20" />
    <p:column width="50">
        <p:outputLabel
            value="Celular" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
            value="#{pacienteMB.celular}"
            disabled="#{pacienteMB.habpaciente}"/>
    </p:column>
    <p:column width="50">
        <p:outputLabel
            value="Referencia de Vivienda" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
            value="#{pacienteMB.refvivienda}"
            disabled="#{pacienteMB.habpaciente}"/>
    </p:column>

```

```

        <p:spacer width="50" height="20"
/>

        </h:panelGrid>
</p:panel>

<p:panel width="80%" header="Condicion
Ocupacion de La Vivienda">
    <h:panelGrid columns="5"
cellpadding="5">
        <p:column width="50">
            <p:outputLabel for="co"
value="Estado de La Vivienda" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu id="co"
value="#{pacienteMB.estvivienda}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="cov"
value="#{pacienteMB.lstvivocup}"
itemLabel="#{cov.descripcion}"
itemValue="#{cov.descripcion}" />
        </p:selectOneMenu>
    </p:column>
    <p:spacer width="50" height="20"
/>

    </h:panelGrid>
</p:panel>

<p:panel width="80%" header="Datos de La
Vivienda">
    <h:panelGrid columns="5"
cellpadding="5">
        <p:column width="50">
            <p:outputLabel for="tv"
value="Tipo de Vivienda" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu id="tv"
value="#{pacienteMB.tipvivienda}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="tipv"
value="#{pacienteMB.lstvivtip}"
itemLabel="#{tipv.descripcion}"
itemValue="#{tipv.descripcion}" />
        </p:selectOneMenu>

```

```

        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel for="acc"

value="Acceso Principal a La Vivienda" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu id="acc"
value="#{pacienteMB.accvivienda}" style="width:350px"
            effect="fold"
            <p:selectItem itemLabel="Seleccione
editable="true" disabled="#{pacienteMB.habpaciente}">
            <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="acpv"
value="#{pacienteMB.Lstvivacc}"
            itemLabel="#{acpv.descripcion}"
            itemValue="#{acpv.descripcion}"
            />
        </p:selectOneMenu>
        </p:column>

        <p:column width="50">
            <p:outputLabel for="tcv"

value="Techo de La Vivienda" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu id="tcv"
value="#{pacienteMB.techovivienda}" style="width:350px"
            effect="fold"
            <p:selectItem itemLabel="Seleccione
editable="true" disabled="#{pacienteMB.habpaciente}">
            <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="tecv"
value="#{pacienteMB.Lstvivtec}"
            itemLabel="#{tecv.descripcion}"
            itemValue="#{tecv.descripcion}" />
        </p:selectOneMenu>
        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel for="pv"

value="Piso de La Vivienda"
            />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu id="pv"
value="#{pacienteMB.pisovivienda}" style="width:350px"
            effect="fold"
            <p:selectItem itemLabel="Seleccione
editable="true" disabled="#{pacienteMB.habpaciente}">
            <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />

```

```

value="#{pacienteMB.Lstvivpis}"
itemLabel="#{pisv.descripcion}"
itemValue="#{pisv.descripcion}"/>
</p:selectOneMenu>
</p:column>
<p:column width="50">
  <p:outputLabel
for="pardv" value="Paredes de La Vivienda" />
</p:column>
<p:column width="50">
  <p:selectOneMenu
id="pardv" value="#{pacienteMB.paredvivienda}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
    <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
    <f:selectItems var="parv"
value="#{pacienteMB.Lstvivpar}"
itemLabel="#{parv.descripcion}"
itemValue="#{parv.descripcion}" />
  </p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
  <p:outputLabel for="etv"
value="Estado del Techo de La Vivienda" />
</p:column>
<p:column width="50">
  <p:selectOneMenu id="etv"
value="#{pacienteMB.esttecvivienda}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
    <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
    <f:selectItems var="estv"
value="#{pacienteMB.Lstvivest}"
itemLabel="#{estv.descripcion}"
itemValue="#{estv.descripcion}"/>
  </p:selectOneMenu>
</p:column>
<p:column width="50">
  <p:outputLabel for="epv"
value="Estado del Piso de La Vivienda" />
</p:column>
<p:column width="50">
  <p:selectOneMenu id="epv"
value="#{pacienteMB.estpisovivienda}" style="width:350px"

```

```

editable="true" disabled="{pacienteMB.habpaciente}"
effect="fold"
<f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
value="{pacienteMB.lstvinvest}"
itemLabel="{espv.descripcion}"
itemValue="{espv.descripcion}" />
</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
<p:outputLabel for="epav"
value="Estado de La Pared de La Vivienda" />
</p:column>
<p:column width="50">
<p:selectOneMenu
id="epav" value="{pacienteMB.estparedvivienda}" style="width:350px"
effect="fold"
editable="true" disabled="{pacienteMB.habpaciente}"
<f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
value="{pacienteMB.lstvinvest}"
itemLabel="{espav.descripcion}"
itemValue="{espav.descripcion}"/>
</p:selectOneMenu>
</p:column>
</h:panelGrid>
</p:panel>
<p:panel width="80%" header="Datos de Hogar
de La Vivienda">
<h:panelGrid columns="5"
cellpadding="5">
<p:column width="50">
<p:outputLabel for="eov"
value="Estado de ocupacion de La Vivienda" />
</p:column>
<p:column width="50">
<p:selectOneMenu id="eov"
value="{pacienteMB.estocupvivienda}" style="width:350px"
effect="fold"
editable="true" disabled="{pacienteMB.habpaciente}"
<f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
value="{pacienteMB.lstvividhogar}"
itemLabel="{esov.descripcion}"
itemValue="{esov.descripcion}"

```

```

        </p:selectOneMenu>
    </p:column>
    <p:spacer width="50" height="20"
/>

    <p:column width="50">
        <p:outputLabel
value="Cantidad de Cuartos en el Hogar" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
value="#{pacienteMB.cantcuartos}"
        disabled="#{pacienteMB.habpaciente}"/>
    </p:column>

    <p:column width="50">
        <p:outputLabel
value="Cantidad de Dormitorios en el Hogar" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
value="#{pacienteMB.cantdormitorio}"
        disabled="#{pacienteMB.habpaciente}"/>
    </p:column>
    <p:spacer width="50" height="20"
/>

    <p:column width="50">
        <p:outputLabel for="pav"
value="Procedencia del Agua en el Hogar" />
    </p:column>
    <p:column width="50">
        <p:selectOneMenu id="pav"
value="#{pacienteMB.procagua}" style="width:350px"
        effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
        <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
        <f:selectItems var="prav"
value="#{pacienteMB.lstvivagua}"
itemLabel="#{prav.descripcion}"
itemValue="#{prav.descripcion}" />
    </p:selectOneMenu>
    </p:column>

    <p:column width="50">
        <p:outputLabel for="sra"
value="De donde se recibe el agua" />
    </p:column>
    <p:column width="50">
        <p:selectOneMenu id="sra"
value="#{pacienteMB.recibagua}" style="width:350px"
        effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">

```

```

Uno" itemValue="" />
value="#{pacienteMB.lstvivagrec}"
itemLabel="#{srav.descripcion}"
itemValue="#{srav.descripcion}" />
/>

value="Tratamiento del agua para beber" />

value="#{pacienteMB.tratagua}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}"
Uno" itemValue="" />
value="#{pacienteMB.lstvivagtrat}"
itemLabel="#{tabv.descripcion}"
itemValue="#{tabv.descripcion}" />

value="Tipo de Servicio Higienico que cuenta en el Hogar" />

value="#{pacienteMB.tipservhig}" style="width:350px"
effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}"
Uno" itemValue="" />
value="#{pacienteMB.lstvivsh}"
itemLabel="#{tshv.descripcion}"
itemValue="#{tshv.descripcion}" />
/>

value="Ubicacion del Servicio Higienico" />

```

```

<f:selectItem itemLabel="Seleccione
<f:selectItems var="srav"

</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"

<p:column width="50">
<p:outputLabel for="tab"
</p:column>
<p:column width="50">
<p:selectOneMenu id="tab"
effect="fold"
<f:selectItem itemLabel="Seleccione
<f:selectItems var="tabv"

</p:selectOneMenu>
</p:column>
<p:column width="50">
<p:outputLabel for="tsh"
</p:column>
<p:column width="50">
<p:selectOneMenu id="tsh"
effect="fold"
<f:selectItem itemLabel="Seleccione
<f:selectItems var="tshv"

</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"

<p:column width="50">
<p:outputLabel for="ush"
</p:column>

```

```

        <p:column width="50">
            <p:selectOneMenu id="ush"
value="#{pacienteMB.ubicservhig}" style="width:350px"
                effect="fold"
            >
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="ushv"
value="#{pacienteMB.lstvivshubic}"
            itemLabel="#{ushv.descripcion}"
            itemValue="#{ushv.descripcion}" />
        </p:selectOneMenu>
    </p:column>

    <p:column width="50">
        <p:outputLabel for="sdu"
value="Servicio de ducha que tiene en el hogar" />
    </p:column>
    <p:column width="50">
        <p:selectOneMenu id="sdu"
value="#{pacienteMB.servducha}" style="width:350px"
                effect="fold"
            >
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="sduv"
value="#{pacienteMB.lstvivsd}"
            itemLabel="#{sduv.descripcion}"
            itemValue="#{sduv.descripcion}" />
        </p:selectOneMenu>
    </p:column>
    <p:spacer width="50" height="20"
/>

    <p:column width="50">
        <p:outputLabel for="elb"
value="Eliminacion de La Basura " />
    </p:column>
    <p:column width="50">
        <p:selectOneMenu id="elb"
value="#{pacienteMB.elimbasm}" style="width:350px"
                effect="fold"
            >
                <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="elbv"
value="#{pacienteMB.lstvivbas}"
            itemLabel="#{elbv.descripcion}"
            itemValue="#{elbv.descripcion}" />
        </p:selectOneMenu>
    </p:column>
    <p:column width="50">

```

```

value="Tipo de Alumbrado" />
<p:outputLabel for="tpa"
</p:column>
<p:column width="50">
  <p:selectOneMenu id="tpa"
    effect="fold"
  >
    <f:selectItem itemLabel="Seleccione
  Uno" itemValue="" />
  <f:selectItems var="tpav"
value="#{pacienteMB.lstvivalum}"
itemLabel="#{tpav.descripcion}"
itemValue="#{tpav.descripcion}" />
</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
  <p:outputLabel for="tpcc"
</p:column>
<p:column width="50">
  <p:selectOneMenu
id="tpcc" value="#{pacienteMB.tipcocina}" style="width:350px"
  effect="fold"
  >
    <f:selectItem itemLabel="Seleccione
  Uno" itemValue="" />
    <f:selectItems var="tpccv"
value="#{pacienteMB.lstvivcoc}"
itemLabel="#{tpccv.descripcion}"
itemValue="#{tpccv.descripcion}" />
</p:selectOneMenu>
</p:column>
<p:column width="50">
  <p:outputLabel for="tipt"
value="Tipo de Transporte" />
</p:column>
<p:column width="50">
  <p:selectOneMenu
id="tipt" value="#{pacienteMB.tiptransp}" style="width:350px"
  effect="fold"
  >
    <f:selectItem itemLabel="Seleccione
  Uno" itemValue="" />
    <f:selectItems var="tptv"
value="#{pacienteMB.lstvivantran}"
itemLabel="#{tptv.descripcion}"
itemValue="#{tptv.descripcion}" />
</p:selectOneMenu>

```

```

        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel for="tmdp"
value="Tiempo de Desplazamiento" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu
id="tmdp" value="#{pacienteMB.tiempdespl}" style="width:350px"
            effect="fold"
            <f:selectItem itemLabel="Seleccione
editable="true" disabled="#{pacienteMB.habpaciente}"
Uno" itemValue="" />
            <f:selectItems var="tmdpv"
value="#{pacienteMB.lstvivtiempo}"
            itemLabel="#{tmdpv.descripcion}"
            itemValue="#{tmdpv.descripcion}" />
        </p:selectOneMenu>
        </p:column>
    </h:panelGrid>
</p:panel>

    <p:panel width="80%" header="Factores de
Vulnerabilidad de La Vivienda">
        <h:panelGrid columns="5"
cellpadding="5">
            <p:column width="50">
                <p:outputLabel for="rmed"
value="Riesgos Medioambientales" />
            </p:column>
            <p:column width="50">
                <p:selectOneMenu
id="rmed" value="#{pacienteMB.rmedioambvivienda}" style="width:350px"
                effect="fold"
                <f:selectItem itemLabel="Seleccione
editable="true" disabled="#{pacienteMB.habpaciente}"
Uno" itemValue="" />
                <f:selectItems var="rmedv"
value="#{pacienteMB.lstrmedioamb}"
                itemLabel="#{rmedv.descripcion}"
                itemValue="#{rmedv.descripcion}" />
            </p:selectOneMenu>
        </p:column>
        <p:spacer width="50" height="20"
/>

            <p:column width="50">
                <p:outputLabel
for="ranim" value="Riesgos de Animales" />
            </p:column>
            <p:column width="50">

```

```

        <p:selectOneMenu
id="ranim" value="#{pacienteMB.ranimal}" style="width:350px"
        effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
        <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
        <f:selectItems var="ranimv"
value="#{pacienteMB.lstranim}"
itemLabel="#{ranimv.descripcion}"
itemValue="#{ranimv.descripcion}" />
        </p:selectOneMenu>
    </p:column>

    <p:column width="50">
        <p:outputLabel
for="riesan" value="Riesgos Sanitarios" />
    </p:column>
    <p:column width="50">
        <p:selectOneMenu
id="riesan" value="#{pacienteMB.rsanitario}" style="width:350px"
        effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
        <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
        <f:selectItems var="riesanv"
value="#{pacienteMB.lstrsanitario}"
itemLabel="#{riesanv.descripcion}"
itemValue="#{riesanv.descripcion}" />
        </p:selectOneMenu>
    </p:column>
    <p:spacer width="50" height="20"
/>

    </h:panelGrid>
</p:panel>

<p:panel width="80%" header="Control">
    <h:panelGrid columns="5"
cellpadding="5">
        <p:column width="50">
            <p:outputLabel
for="contentrev" value="Control de Entrevistas Efectivas" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu
id="contentrev" value="#{pacienteMB.contentr}" style="width:350px"
            effect="fold"
editable="true" disabled="#{pacienteMB.habpaciente}">
            <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="rmedv"
value="#{pacienteMB.lstcontentr}"
itemLabel="#{rmedv.descripcion}"

```

```

itemValue="#{rmedv.descripcion}" />
</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"
/>

<p:column width="50">
<p:outputLabel
value="Control de Trabajo: Nombre del Encuestador" />
</p:column>
<p:column width="50">
<p:inputText size="50"

value="#{pacienteMB.nombenc}"

disabled="#{pacienteMB.habpaciente}"/>
</p:column>

<p:column width="50">
<p:outputLabel
value="Control de Trabajo: Codigo del Encuestador" />
</p:column>
<p:column width="50">
<p:inputText size="50"

value="#{pacienteMB.codenc}"

disabled="#{pacienteMB.habpaciente}"/>
</p:column>
<p:spacer width="50" height="20"
/>

<p:column width="50">
<p:outputLabel
for="popup" value="Fecha de ultima visita" />
</p:column>
<p:column width="50">
<p:calendar id="popup"

value="#{pacienteMB.fecha}"

disabled="#{pacienteMB.habpaciente}"/>
</p:column>

<p:column width="50">
<p:outputLabel
value="Numero de Visita" />

</p:column>
<p:column width="50">
<p:inputText size="50"

value="#{pacienteMB.numvis}"

disabled="#{pacienteMB.habpaciente}"/>
</p:column>

</h:panelGrid>

```

```

        </p:panel>

        <p:commandButton value="Guardar"
actionListener="#{pacienteMB.guardar}"
disabled="#{pacienteMB.habpaciente}"
update="growl form1">
        </p:commandButton>

    </h:panelGrid>
</div>

</h:form>

</ui:define>
</ui:composition>

</html>

```

Vista Mortalidad Familiar

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xmlns:ui="http://java.sun.com/jsf/facelets"
xmlns:f="http://java.sun.com/jsf/core"
xmlns:h="http://java.sun.com/jsf/html"
xmlns:p="http://primefaces.org/ui">

    <ui:composition template="/WEB-INF/template.xhtml">
        <ui:define name="title">FICHA FAMILIAR</ui:define>
        <ui:define name="content">
            <h:form id="form1">
                <p:growl id="growl" showDetail="true" showSummary="true"
                    sticky="true" />
                <div align="left">

                    <h:panelGrid columns="4" cellpadding="5">
                        <p:inputText id="datosConsulta"
size="30"
value="#{mortalidadMB.cedula}" />

                        <p:watermark for="datosConsulta"
value="Buscar por
identificacion" />

                        <p:commandButton
value="Consultar" id="btnConsultar"

actionListener="#{mortalidadMB.buscarPaciente}" update="form1" />

                        <p:commandButton value="Limpiar"
id="btnLimpiar"

actionListener="#{mortalidadMB.Limpiar}" update="form1" />
                    </h:panelGrid>

                    <h:panelGrid columns="1" cellpadding="5" width="100%">

```

```

<p:panel width="80%" header="Mortalidad Familiar">
  <h:panelGrid columns="5" cellpadding="5">
    <p:column width="50">
      <p:outputLabel
value="Nombre de Familiar" />
    </p:column>
    <p:column width="50">
      <p:inputText size="50"
value="#{mortalidadMB.nombre}"
disabled="#{mortalidadMB.habpaciente}"/>
    </p:column>
    <p:spacer width="50" height="20" />
    <p:column width="50">
      <p:outputLabel
value="Apellido de Familiar" />
    </p:column>
    <p:column width="50">
      <p:inputText size="50"
value="#{mortalidadMB.apellido}"
disabled="#{mortalidadMB.habpaciente}"/>
    </p:column>
    <p:column width="50">
      <p:outputLabel for="cd"
value="Sexo de Familiar" />
    </p:column>
    <p:column width="50">
      <p:selectOneMenu id="cd"
value="#{mortalidadMB.sexo}" style="width:340px"
effect="fold"
editable="true" disabled="#{mortalidadMB.habpaciente}">
        <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
        <f:selectItems var="cdaa"
value="#{mortalidadMB.lstsexo}"
itemLabel="#{cdaa.descripcion}"
itemValue="#{cdaa.descripcion}"/>
      </p:selectOneMenu>
    </p:column>
    <p:spacer width="50" height="20" />
    <p:column>
      <p:outputLabel
value="Edad de Familiar" />
    </p:column>
    <p:column>
      <p:inputText size="50"

```

```

value="#{mortalidadMB.edad}" disabled="#{mortalidadMB.habpaciente}"/>
</p:column>

<p:column>
<p:outputLabel for="cf"
value="Documentacion de Familiar" />
</p:column>

<p:column>
<p:selectOneMenu id="cf"
value="#{mortalidadMB.document}" style="width:340px"
effect="fold"
editable="true" disabled="#{mortalidadMB.habpaciente}">
<f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
<f:selectItems var="cdab"
value="#{mortalidadMB.lstdocum}"
itemLabel="#{cdab.descripcion}"
itemValue="#{cdab.descripcion}"/>
</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"
/>

<p:column>
<p:outputLabel
value="Identificacion Familiar" />
</p:column>
<p:column>
<p:inputText size="50"
value="#{mortalidadMB.identif}"
disabled="#{mortalidadMB.habpaciente}"/>
</p:column>

<p:column>
<p:outputLabel for="cg"
value="Parentesco Familiar (Con relacion al jefe del hogar)" />
</p:column>

<p:column>
<p:selectOneMenu id="cg"
value="#{mortalidadMB.parent}" style="width:340px"
effect="fold"
editable="true" disabled="mortalidadMB.habpaciente">
<f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
<f:selectItems var="cdac"
value="#{mortalidadMB.lstparen}"
itemLabel="#{cdac.descripcion}"
itemValue="#{cdac.descripcion}"/>
</p:selectOneMenu>
</p:column>

```

```

/>
<p:spacer width="50" height="20"

<p:column>
  <p:outputLabel
for="popup" value="Fecha Muerte de Familiar" />
</p:column>
<p:column>
  <p:calendar id="popup"
value="#{mortalidadMB.fecha}" disabled="#{mortalidadMB.habpaciente}"/>
</p:column>

<p:column>
  <p:outputLabel for="ch"
value="Clase de muerte del Familiar" />
</p:column>

<p:column>
  <p:selectOneMenu id="ch"
value="#{mortalidadMB.clase}" style="width:340px"
effect="fold"
editable="true" disabled="#{mortalidadMB.habpaciente}">
    <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
    <f:selectItems var="cdad"
value="#{mortalidadMB.lstclase}"
itemLabel="#{cdad.descripcion}"
itemValue="#{cdad.descripcion}"/>
  </p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"

/>

<p:column>
  <p:outputLabel
value="Causa de La Muerte de Familiar" />
</p:column>
<p:column>
  <p:inputText size="50"
value="#{mortalidadMB.causa}" disabled="#{mortalidadMB.habpaciente}"/>
</p:column>

<p:column>
  <p:outputLabel
value="Observacion" />
</p:column>
<p:column>
  <p:inputText size="50"
value="#{mortalidadMB.observacion}"
disabled="#{mortalidadMB.habpaciente}"/>
</p:column>

```

```

/>
                                <p:spacer width="50" height="20"

                                </h:panelGrid>
                                </p:panel>

                                <p:commandButton value="Guardar"
actionListener="#{mortalidadMB.guardar()}"
disabled="#{mortalidadMB.habpaciente}"
                                update="growl
form1"></p:commandButton>

                                </h:panelGrid>
                                </div>

                                </h:form>
                                </ui:define>
</ui:composition>

</html>

```

Vista Intervención Familiar

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xmlns:ui="http://java.sun.com/jsf/facelets"
xmlns:f="http://java.sun.com/jsf/core"
xmlns:h="http://java.sun.com/jsf/html"
xmlns:p="http://primefaces.org/ui">

<ui:composition template="/WEB-INF/template.xhtml">
<ui:define name="title">FICHA FAMILIAR</ui:define>
<ui:define name="content">
<h:form id="form1">
<p:growl id="growl" showDetail="true" showSummary="true"
sticky="true" />
<div align="left">

                                <h:panelGrid columns="4" cellpadding="5">
                                <p:inputText id="datosConsulta"
size="30"
value="#{intervencionMB.cedula}" />

                                <p:watermark for="datosConsulta"
value="Buscar por
identificacion" />

                                <p:commandButton
value="Consultar" id="btnConsultar"

                                actionListener="#{intervencionMB.buscarPaciente}" update="form1" />
                                <p:commandButton value="Limpiar"
id="btnLimpiar"

                                actionListener="#{intervencionMB.Limpiar}" update="form1" />

```

```

        </h:panelGrid>

        <h:panelGrid columns="1" cellpadding="5" width="100%">
            <p:panel width="80%" header="Evaluacion de
Las Intervenciones Familiar (Registre de acuerdo a Los codigos)">
                <h:panelGrid columns="5"
cellpadding="5">

                    <p:column width="50">
                        <p:outputLabel
value="Identificacion" />

                    </p:column>
                    <p:column width="50">
                        <p:inputText size="50"

value="#{intervencionMB.identif}"

disabled="#{intervencionMB.habpaciente}"/>
                    </p:column>
                    <p:spacer width="50" height="20"
/>

                    <p:column width="50">
                        <p:outputLabel
value="Nombre Completo" />

                    </p:column>
                    <p:column width="50">
                        <p:inputText size="50"

value="#{intervencionMB.nombre}"

disabled="#{intervencionMB.habpaciente}"/>
                    </p:column>

                    <p:column width="50">
                        <p:outputLabel
for="popup" value="Fecha de Registro" />

                    </p:column>
                    <p:column width="50">
                        <p:calendar id="popup"

value="#{intervencionMB.fecha}"

disabled="#{intervencionMB.habpaciente}"/>
                    </p:column>
                    <p:spacer width="50" height="20"
/>

                    <p:column width="50">
                        <p:outputLabel
value="Codigo Cumplimiento" />

                    </p:column>
                    <p:column width="50">
                        <p:inputText size="50"

value="#{intervencionMB.cumplimiento}"

disabled="#{intervencionMB.habpaciente}"/>

```

```

        </p:column>

        <p:column width="50">
            <p:outputLabel
value="Codigo de Resultados" />
        </p:column>

        <p:column width="50">
            <p:inputText size="50"
value="#{intervencionMB.resultado}"
disabled="#{intervencionMB.habpaciente}"/>
        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel
value="Causas y Dificultades" />
        </p:column>
        <p:column width="50">
            <p:inputText size="50"
value="#{intervencionMB.dificultades}"
disabled="#{intervencionMB.habpaciente}"/>
        </p:column>

        <p:column width="50">
            <p:outputLabel
value="Detalle de Los Acuerdos alcanzados" />
        </p:column>
        <p:column width="50">
            <p:inputText size="50"
value="#{intervencionMB.acuerdos}"
disabled="#{intervencionMB.habpaciente}"/>
        </p:column>
        <p:spacer width="50" height="20"
/>

        </h:panelGrid>
    </p:panel>

    <p:commandButton value="Guardar"
actionListener="#{intervencionMB.guardar()}"
disabled="#{intervencionMB.habpaciente}"
update="growl
form1"></p:commandButton>

    </h:panelGrid>
</div>

</h:form>

```

```

    </ui:define>
</ui:composition>

</html>

```

Vista Diagnostico Familiar

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xmlns:ui="http://java.sun.com/jsf/facelets"
  xmlns:f="http://java.sun.com/jsf/core"
  xmlns:h="http://java.sun.com/jsf/html"
  xmlns:p="http://primefaces.org/ui">

  <ui:composition template="/WEB-INF/template.xhtml">
    <ui:define name="title">FICHA FAMILIAR</ui:define>
    <ui:define name="content">
      <h:form id="form1">
        <p:growl id="growl" showDetail="true" showSummary="true"
          sticky="true" />
        <div align="left">

          <h:panelGrid columns="4" cellpadding="5">
            <p:inputText id="datosConsulta"
              size="30"
              value="#{diagnosticoMB.cedula}" />
            <p:watermark for="datosConsulta"
              value="Buscar por
              identificacion" />
            <p:commandButton
              value="Consultar" id="btnConsultar"
              actionListener="#{diagnosticoMB.buscarPaciente}" update="form1" />
            <p:commandButton value="Limpiar"
              id="btnLimpiar"
              actionListener="#{diagnosticoMB.limpiar}" update="form1" />
          </h:panelGrid>

          <h:panelGrid columns="1" cellpadding="5" width="100%">
            <p:panel width="80%" header="Actividad de Diagnostico
              Familiar">
              <h:panelGrid columns="1" cellpadding="5">
                <p:panel width="80%" header="Visita de
                  Diagnostico">
                    <h:panelGrid columns="5"
                      cellpadding="5">
                      <p:column width="50">
                        <p:outputLabel
                          value="Identificacion" />
                      </p:column>
                      <p:column width="50">

```

```

        <p:inputText size="50"
value="#{diagnosticoMB.identif}"
disabled="#{diagnosticoMB.habpaciente}"/>
    </p:column>
    <p:spacer width="50" height="20"
/>

    <p:column width="50">
        <p:outputLabel
value="Nombre Completo" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"
value="#{diagnosticoMB.nombre}"
disabled="#{diagnosticoMB.habpaciente}"/>
    </p:column>

    <p:column width="50">
        <p:outputLabel for="cd"
value="Clasificacion Diagnostica" />
    </p:column>
    <p:column width="50">
        <p:selectOneMenu id="cd"
value="#{diagnosticoMB.clasif}" style="width:340px"
effect="fold"
editable="true" disabled="#{diagnosticoMB.habpaciente}">
            <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
            <f:selectItems var="cdaa"
value="#{diagnosticoMB.lstclasificacion}"
itemLabel="#{cdaa.descripcion}"
itemValue="#{cdaa.descripcion}"/>
        </p:selectOneMenu>
    </p:column>
    <p:spacer width="50" height="20"
/>

    <p:column width="50">
        <p:outputLabel
for="popup" value="Fecha de Diagnostico" />
    </p:column>
    <p:column width="50">
        <p:calendar id="popup"
value="#{diagnosticoMB.fecha}"
disabled="#{diagnosticoMB.habpaciente}"/>
    </p:column>
    <p:column width="50">

```

```

value="Describir el Problema Actual" />
        <p:outputLabel
        </p:column>
        <p:column width="50">
            <p:inputText size="50"

            value="#{diagnosticoMB.probl}"
            disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>
    />
    <p:spacer width="50" height="20"

    <p:column width="50">
        <p:outputLabel
    </p:column>
    <p:column width="50">
        <p:inputText size="50"

        value="Nombre Responsable" />
    </p:column>
    <p:column width="50">
        <p:inputText size="50"

        value="#{diagnosticoMB.nombresp}"
        disabled="#{diagnosticoMB.habpaciente}"/>
    </p:column>

    </h:panelGrid>
</p:panel>

<p:panel width="80%" header="Visita de Primer
Diagnostico">
    cellpadding="5">
        <h:panelGrid columns="5"

        <p:column width="50">
            <p:outputLabel
        </p:column>
        <p:column width="50">
            <p:inputText size="50"

            value="Identificacion" />
        </p:column>
        <p:column width="50">
            <p:inputText size="50"

            value="#{diagnosticoMB.identif1}"
            disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>
        <p:spacer width="50" height="20"

        <p:column width="50">
            <p:outputLabel
        </p:column>
        <p:column width="50">
            <p:inputText size="50"

            value="Nombre Completo" />
        </p:column>
        <p:column width="50">
            <p:inputText size="50"

            value="#{diagnosticoMB.nombre1}"
            disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>

```

```

value="Clasificacion Diagnostica" />
<p:column width="50">
  <p:outputLabel for="ce"
</p:column>
<p:column width="50">
  <p:selectOneMenu id="ce"
value="#{diagnosticoMB.clasif1}" style="width:340px"
effect="fold"
editable="true" disabled="#{diagnosticoMB.habpaciente}">
  <f:selectItem itemLabel="Seleccione
Uno" itemValue="" />
  <f:selectItems var="cdab"
value="#{diagnosticoMB.lstclasificacion}"
itemLabel="#{cdab.descripcion}"
itemValue="#{cdab.descripcion}"/>
</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
  <p:outputLabel
for="popup1" value="Fecha de Diagnostico" />
</p:column>
<p:column width="50">
  <p:calendar id="popup1"
value="#{diagnosticoMB.fecha1}"
disabled="#{diagnosticoMB.habpaciente}"/>
</p:column>
<p:column width="50">
  <p:outputLabel
value="Describir el Problema Actual" />
</p:column>
<p:column width="50">
  <p:inputText size="50"
value="#{diagnosticoMB.probl1}"
disabled="#{diagnosticoMB.habpaciente}"/>
</p:column>
<p:spacer width="50" height="20"
/>
<p:column width="50">
  <p:outputLabel
value="Nombre Responsable" />
</p:column>
<p:column width="50">
  <p:inputText size="50"

```

```

        value="#{diagnosticoMB.nombresp1}"
        disabled="#{diagnosticoMB.habpaciente}"/>
    </p:column>

    </h:panelGrid>
</p:panel>

<p:panel width="80%" header="Visita de Segundo
Diagnostico">
    <h:panelGrid columns="5"
        cellpadding="5">

        <p:column width="50">
            <p:outputLabel
                value="Identificacion" />
        </p:column>
        <p:column width="50">
            <p:inputText size="50"
                value="#{diagnosticoMB.identif2}"
                disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>
        <p:spacer width="50" height="20" />

        <p:column width="50">
            <p:outputLabel
                value="Nombre Completo" />
        </p:column>
        <p:column width="50">
            <p:inputText size="50"
                value="#{diagnosticoMB.nombre2}"
                disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>

        <p:column width="50">
            <p:outputLabel for="cf"
                value="Clasificacion Diagnostica" />
        </p:column>
        <p:column width="50">
            <p:selectOneMenu id="cf"
                value="#{diagnosticoMB.clasif2}" style="width:340px"
                effect="fold"
                editable="true" disabled="#{diagnosticoMB.habpaciente}">
                <f:selectItem itemLabel="Seleccione
                Uno" itemValue="" />
                <f:selectItems var="cdac"
                    value="#{diagnosticoMB.lstclasificacion}"
                    itemLabel="#{cdac.descripcion}"
                    itemValue="#{cdac.descripcion}"/>
            </p:selectOneMenu>
        </p:column>
    </h:panelGrid>
</p:panel>

```

```

        </p:selectOneMenu>
        </p:column>
        <p:spacer width="50" height="20"
/>

        <p:column width="50">
            <p:outputLabel
for="popup2" value="Fecha de Diagnostico" />
        </p:column>
        <p:column width="50">
            <p:calendar id="popup2"
                value="#{diagnosticoMB.fecha2}"
                disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>

        <p:column width="50">
            <p:outputLabel
value="Describir el Problema Actual" />
        </p:column>
        <p:column width="50">
            <p:inputText size="50"
                value="#{diagnosticoMB.probl2}"
                disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>

        <p:spacer width="50" height="20"
/>

        <p:column>
            <p:outputLabel
value="Nombre Responsable" />
        </p:column>
        <p:column>
            <p:inputText size="50"
                value="#{diagnosticoMB.nombresp2}"
                disabled="#{diagnosticoMB.habpaciente}"/>
        </p:column>

        </h:panelGrid>
    </p:panel>

    <p:panel width="80%" header="Visita de Tercer
Diagnostico">
        <h:panelGrid columns="5"
cellpadding="5">

            <p:column width="50">
                <p:outputLabel
value="Identificacion" />
            </p:column>
            <p:column width="50">
                <p:inputText size="50"

```

```

        value="#{diagnosticoMB.identif3}"
        disabled="#{diagnosticoMB.habpaciente}"/>
    />

    value="Nombre Completo" />

        value="#{diagnosticoMB.nombre3}"
        disabled="#{diagnosticoMB.habpaciente}"/>

    value="Clasificacion Diagnostica" />

        value="#{diagnosticoMB.clasif3}" style="width:340px"
        effect="fold"
        editable="true" disabled="#{diagnosticoMB.habpaciente}">
        <f:selectItem itemLabel="Seleccione
        Uno" itemValue="" />
        <f:selectItems var="cdad"
        itemLabel="#{cdad.descripcion}"
        itemValue="#{cdad.descripcion}"/>

    />

    for="popup3" value="Fecha de Diagnostico" />

    id="popup3"

        value="#{diagnosticoMB.fecha3}"
        disabled="#{diagnosticoMB.habpaciente}"/>

    value="Describir el Problema Actual" />

```

```

</p:column>
<p:spacer width="50" height="20"

```

```

<p:column width="50">
    <p:outputLabel
</p:column>
<p:column width="50">
    <p:inputText size="50"

```

```

</p:column>

<p:column width="50">
    <p:outputLabel for="cg"
</p:column>

```

```

<p:column width="50">
    <p:selectOneMenu id="cg"
        effect="fold"
        <f:selectItem itemLabel="Seleccione
        Uno" itemValue="" />
        <f:selectItems var="cdad"

```

```

</p:selectOneMenu>
</p:column>
<p:spacer width="50" height="20"

```

```

<p:column width="50">
    <p:outputLabel
</p:column>
<p:column>
    <p:calendar width="50"

```

```

</p:column>

<p:column width="50">
    <p:outputLabel

```

```

        value="#{diagnosticoMB.probl3}"
        disabled="#{diagnosticoMB.habpaciente}"/>
    />

    value="Nombre Responsable" />

    value="#{diagnosticoMB.nombresp3}"
    disabled="#{diagnosticoMB.habpaciente}"/>

    </h:panelGrid>
</p:panel>

    </h:panelGrid>
</p:panel>

    <p:commandButton value="Guardar"
        actionListener="#{diagnosticoMB.guardar()}"
        disabled="#{diagnosticoMB.habpaciente}"
        update="growl
form1"></p:commandButton>

    </h:panelGrid>
</div>

</h:form>

</ui:define>
</ui:composition>

</html>

```

Controlador del Sistema Ficha Familiar

Controlador Datos Paciente

```
package edu.cisc.controlador;
```

```
import java.io.Serializable;
```

```
import java.util.ArrayList;
import java.util.Date;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import javax.annotation.PostConstruct;
import javax.faces.application.Application;
import javax.faces.application.FacesMessage;
import javax.faces.application.ViewHandler;
import javax.faces.bean.SessionScoped;
import javax.faces.component.UIViewRoot;
import javax.faces.context.FacesContext;
import javax.persistence.NoResultException;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

import edu.cisc.dao.FfacantonesDao;
import edu.cisc.dao.FfadistritoDao;
import edu.cisc.dao.FfadivisionDao;
import edu.cisc.dao.FfaedificioDao;
import edu.cisc.dao.FfalocalidadDao;
import edu.cisc.dao.FfamanzanaDao;
import edu.cisc.dao.FfaparroquiaDao;
import edu.cisc.dao.FfasectorDao;
import edu.cisc.dao.Ffavivienda_ubicacionDao;
//import edu.cisc.dao.ParFamiliaDao;
//import edu.cisc.dao.ParPacienteDao;
//import edu.cisc.dao.ParPersonaDao;
import edu.cisc.dao.FfaviviendaaccesoDao;
import edu.cisc.dao.FfaviviendaagrecDao;
import edu.cisc.dao.FfaviviendaagtratDao;
import edu.cisc.dao.FfaviviendaaguaDao;
```

```
import edu.cisc.dao.FfaviviendaalumbradoDao;
import edu.cisc.dao.FfaviviendabasuraDao;
import edu.cisc.dao.FfaviviendacocinaDao;
import edu.cisc.dao.FfaviviendadhogarDao;
import edu.cisc.dao.FfaviviendaestadoDao;
import edu.cisc.dao.FfaviviendahogarDao;
import edu.cisc.dao.FfaviviendanumeroDao;
import edu.cisc.dao.FfaviviendaocupacionDao;
import edu.cisc.dao.FfaviviendaparedesDao;
import edu.cisc.dao.FfaviviendapisoDao;
import edu.cisc.dao.FfaviviendaservduchaDao;
import edu.cisc.dao.FfaviviendashDao;
import edu.cisc.dao.FfaviviendashubicDao;
import edu.cisc.dao.FfaviviendatechoDao;
import edu.cisc.dao.FfaviviendatiempoDao;
import edu.cisc.dao.FfaviviendatiposDao;
import edu.cisc.dao.FfaviviendatransporteDao;
import edu.cisc.dao.FfazonaDao;
//import edu.cisc.dao.AtmsectoreDao;
import edu.cisc.dao.HcflocalizacionDao;
import edu.cisc.dao.ProestablecimientososaludDao;
import edu.cisc.dao.FfaviviendariesgosmedioDao;
import edu.cisc.dao.FfaviviendariesgossanitarioDao;
import edu.cisc.dao.FfaviviendariesgosanimalDao;
import edu.cisc.dao.FfaviviendaentrevistaDao;

import edu.cisc.modelo.Ffacantones;
import edu.cisc.modelo.Ffadistrito;
import edu.cisc.modelo.Ffadivision;
import edu.cisc.modelo.Ffaedificio;
import edu.cisc.modelo.Ffalocalidad;
import edu.cisc.modelo.Ffamanzana;
import edu.cisc.modelo.Ffaparroquia;
import edu.cisc.modelo.Ffasector;
```

```
import edu.cisc.modelo.Ffavivienda_intervencion;
import edu.cisc.modelo.Ffavivienda_ubicacion;
//import edu.cisc.modelo.Parfamilia;
//import edu.cisc.modelo.Parpaciente;
//import edu.cisc.modelo.Parpersona;
import edu.cisc.modelo.Ffaviviendaacceso;
import edu.cisc.modelo.Ffaviviendaagrec;
import edu.cisc.modelo.Ffaviviendaagtrat;
import edu.cisc.modelo.Ffaviviendaagua;
import edu.cisc.modelo.Ffaviviendaalumbrado;
import edu.cisc.modelo.Ffaviviendabasura;
import edu.cisc.modelo.Ffaviviendacocina;
import edu.cisc.modelo.Ffaviviendadhogar;
import edu.cisc.modelo.Ffaviviendaestado;
import edu.cisc.modelo.Ffaviviendahogar;
import edu.cisc.modelo.Ffaviviendanumero;
import edu.cisc.modelo.Ffaviviendaocupacion;
import edu.cisc.modelo.Ffaviviendaparedes;
import edu.cisc.modelo.Ffaviviendapiso;
import edu.cisc.modelo.Ffaviviendaservducha;
import edu.cisc.modelo.Ffaviviendash;
import edu.cisc.modelo.Ffaviviendashubic;
import edu.cisc.modelo.Ffaviviendatecho;
import edu.cisc.modelo.Ffaviviendatiempo;
import edu.cisc.modelo.Ffaviviendatipos;
import edu.cisc.modelo.Ffaviviendatransporte;
import edu.cisc.modelo.Ffazona;
//import edu.cisc.modelo.Atmsectore;
import edu.cisc.modelo.HcfLocalizacion;
import edu.cisc.modelo.ProEstablecimientoSalud;
import edu.cisc.modelo.Ffaviviendariesgosmedio;
import edu.cisc.modelo.Ffaviviendariesgossanitario;
import edu.cisc.modelo.Ffaviviendariesgosanimal;
import edu.cisc.modelo.Ffaviviendaentrevista;
```

```

@Component(value="pacienteMB")
@SessionScoped
public class DatosPacienteMB implements Serializable{

    private static final long serialVersionUID = 1L;

    //@Autowired(required=true)

    //private Parpersona nuevaPersona;
    //private Parfamilia nuevaFamilia;
    //private Atmsectore nuevoSector;
    private HcfLocalizacion local;
    private Ffparroquia parr;
    private Ffadistrito dist;
    private Ffazona zon;
    private Ffalocalidad loc;
    private Ffasector sex;
    private Ffamanzana manz;
    private Ffaedificio edi;
    private Ffaviviendanumero vnum;
    private Ffaviviendahogar vhog;
    private Ffaviviendaocupacion vestocup;
    private Ffaviviendatipos vtipos;
    private Ffaviviendaacceso vacceso;
    private Ffaviviendatecho vtecho;
    private Ffaviviendapiso vpiso;
    private Ffaviviendaparedes vparedes;
    private Ffaviviendaestado vestado;
    private Ffaviviendahogar vdhog;
    private Ffaviviendaagua vagua;
    private Ffaviviendaagrec vagrec;

```

```
private Ffaviviendaagrec vagtrat;
private Ffaviviendash vsh;
private Ffaviviendashubic vshubic;
private Ffaviviendaservducha vserduc;
private Ffaviviendabasura vbasura;
private Ffaviviendaalumbrado valumbrado;
private Ffaviviendacocina vcocina;
private Ffaviviendatiempo vtiempo;
private Ffaviviendatransporte vtrans;
private Ffavivienda_ubicacion vtabguardar;
private ProEstablecimientoSalud estab;
private Ffacantones cantones;
private Ffdivision div;
//private Parpaciente nPaciente;
```

```
private String ident;//0
private String pnomb;//1
private String snomb;//2
private String papell;//3
private String sapell;//4
private String establecimiento;//5
private String provinc;//6
private String canton;//7
private String parroquia;//8
private String distrito;//9
private String zona;//10
private String localidad;//11
private String sector;//12
private String manzana;//13
private String division;//14
private String edificio;//15
private String numvivienda;//16
private String numhogar;//17
```

private String pcalle;//18
private String scalle;//19
private String numcasa;//20
private String conjhabitac;//21
private String numlote;//22
private String numdepart;//23
private String numpiso;//24
private String telefono;//25
private String celular;//26
private String refvivienda;//27
private String estvivienda;//28
private String tipvivienda;//29
private String accvivienda;//30
private String techovivienda;//31
private String pisovivienda;//32
private String paredvivienda;//33
private String esttecovivienda;//34
private String estpisovivienda;//35
private String estparedvivienda;//36
private String estocupvivienda;//37
private String cantcuartos;//38
private String cantdormitorio;//39
private String procagua;//40
private String recibagua;//41
private String tratagua;//42
private String tipservhig;//43
private String ubicservhig;//44
private String servducha;//45
private String elimbas;//46
private String tipalumb;//47
private String tipcocina;//48
private String tiptransp;//49
private String tiempdespl;//50
private Date fechausu;//51

```

private Date fechamod;//52
private String codusu;//53
private String codmod;//54
private String estado;//55
private String rmedioambvivienda;//56
private String ranimal;//57
private String rsanitario;//58
private String contentr;//59
private String nombenc;//60
private String codenc;//61
private Date fecha;//62
private String numvis;//62
//para seleccionar la cedula
private String cedula;
//para validar que se visualice los botones
private boolean habpaciente=false;

private Map<String,String> establec = new HashMap<String, String>();

//@Autowired
//AtmsectoreDao sectorDao;

private List<HcfLocalizacion> IstLocalizacion;
private List<Ffaparroquia> Istparroquia;
private List<Ffacantones> IstLocalizacionCanton;
private List<ProEstablecimientoSalud> IstLocalizacionEstab;
private List<Ffadistrito> Istdistrito;
private List<Ffazona> Istzona;
private List<Ffalocalidad> Istlocal;
private List<Ffasector> Istsector;
private List<Ffamanzana> Istmanzana;
private List<Ffadivision> Istdivision;
private List<Ffaedificio> Istedificio;

```

```
private List<Ffaviviendanumero> Istvivnumero;
private List<Ffaviviendahogar> Istvivhogar;
private List<Ffaviviendaocupacion> Istvivocup;
private List<Ffaviviendatipos> Istvivtip;
private List<Ffaviviendaacceso> Istvivacc;
private List<Ffaviviendatecho> Istvivtec;
private List<Ffaviviendapiso> Istvivpis;
private List<Ffaviviendaparedes> Istvivpar;
private List<Ffaviviendaestado> Istvivest;
private List<Ffaviviendadhogar> Istvividhogar;
private List<Ffaviviendaagua> Istvivagua;
private List<Ffaviviendaagrec> Istvivagrec;
private List<Ffaviviendaagtrat> Istvivagtrat;
private List<Ffaviviendash> Istvivsh;
private List<Ffaviviendashubic> Istvivshubic;
private List<Ffaviviendaservducha> Istvivsd;
private List<Ffaviviendabasura> Istvivbas;
private List<Ffaviviendaalumbrado> Istvivalum;
private List<Ffaviviendacocina> Istvivcoc;
private List<Ffaviviendatransporte> Istvivtran;
private List<Ffaviviendatiempo> Istvivtiempo;
private List<Ffaviviendariesgosmedio> Istrmedioamb;
private List<Ffaviviendariesgosanimal> Istranim;
private List<Ffaviviendariesgossanitario> Istrsanitario;
private List<Ffaviviendaentrevista> Istcontentr;
```

```
@Autowired
```

```
private HcflocalizacionDao localizacionDao;
```

```
@Autowired
```

```
private FfacantonesDao cantonesDao;
```

```
@Autowired
```

```
private ProestablecimientosaludDao establecimientosaludDao;
```

```
@Autowired
```

```
private FfparroquiaDao parrDao;
```

```
@Autowired
private FfadrastroDao distDao;
@Autowired
private FfalocalidadDao locDao;
@Autowired
private FfasectorDao secDao;
@Autowired
private FfazonaDao zonaDao;
@Autowired
private FfamanzanaDao manDao;
@Autowired
private FfdivisionDao divDao;
@Autowired
private FfaedificioDao ediDao;
@Autowired
private FfaviviendanumeroDao vnumDao;
@Autowired
private FfaviviendahogarDao vhogDao;
@Autowired
private FfaviviendaocupacionDao vocupDao;
@Autowired
private FfaviviendatiposDao vtipDao;
@Autowired
private FfaviviendaaccesoDao vaccDao;
@Autowired
private FfaviviendatechoDao vtecDao;
@Autowired
private FfaviviendapisoDao vpisDao;
@Autowired
private FfaviviendaparedesDao vparDao;
@Autowired
private FfaviviendaestadoDao vestDao;
@Autowired
private FfaviviendadhogarDao vdhogDao;
```

```
@Autowired
private FfaviviendaaguaDao vagDao;
@Autowired
private FfaviviendaagrecDao vagrecDao;
@Autowired
private FfaviviendaagtratDao vagtratDao;
@Autowired
private FfaviviendashDao vshDao;
@Autowired
private FfaviviendashubicDao vshubicDao;
@Autowired
private FfaviviendaservduchaDao vsdDao;
@Autowired
private FfaviviendabasuraDao vbasDao;
@Autowired
private FfaviviendaalumbradoDao valumDao;
@Autowired
private FfaviviendacocinaDao vcocDao;
@Autowired
private FfaviviendatransporteDao vtransDao;
@Autowired
private FfaviviendatiempoDao vtiempoDao;
@Autowired
private FfaviviendariesgosmedioDao vrmedDao;
@Autowired
private FfaviviendariesgosanimalDao vraniDao;
@Autowired
private FfaviviendariesgossanitarioDao vrsanDao;
@Autowired
private FfaviviendaentrevistaDao ventrevDao;
@Autowired
private Ffavivienda_ubicacionDao vguardar;
/*@Autowired
ParFamiliaDao familiaDao;
```

```

    @Autowired
    ParPacienteDao pacienteDao;*/
    //@Autowired
    //ParPersonaDao personaDao;

    @PostConstruct
    public void init() {
        //refresh();
        IstLocalizacion = new ArrayList<HcfLocalizacion>();
        IstLocalizacionCanton = new ArrayList<Ffacantones>();
        IstLocalizacionEstab = new
ArrayList<ProEstablecimientoSalud>();
        Istparroquia = new ArrayList<Ffaparroquia>();
        Istdistrito = new ArrayList<Ffadistrito>();
        Istzona = new ArrayList<Ffazona>();
        Istlocal = new ArrayList<Ffalocalidad>();
        Istsector = new ArrayList<Ffasector>();
        Istmanzana = new ArrayList<Ffamanzana>();
        Istdivision = new ArrayList<Ffadivision>();
        Istedificio = new ArrayList<Ffaedificio>();
        Istvivnumero = new ArrayList<Ffaviviendanumero>();
        Istvivhogar = new ArrayList<Ffaviviendahogar>();
        Istvivocup = new ArrayList<Ffaviviendaocupacion>();
        Istvivtip = new ArrayList<Ffaviviendatipos>();
        Istvivacc = new ArrayList<Ffaviviendaacceso>();
        Istvivtec = new ArrayList<Ffavivientecho>();
        Istvivpis = new ArrayList<Ffaviviendapiso>();
        Istvivpar = new ArrayList<Ffaviviendaparedes>();
        Istvivest = new ArrayList<Ffaviviendaestado>();
        Istvivdhogar = new ArrayList<Ffaviviendadhogar>();
        Istvivagua = new ArrayList<Ffaviviendaagua>();
        Istvivagrec = new ArrayList<Ffaviviendaagrec>();
        Istvivagtrat = new ArrayList<Ffaviviendaagtrat>();
    }

```

```
Istvivsh = new ArrayList<Ffaviviendash>();  
Istvivshubic = new ArrayList<Ffaviviendashubic>();  
Istvivsd = new ArrayList<Ffaviviendaservducha>();  
Istvivbas = new ArrayList<Ffaviviendabasura>();  
Istvivalum = new ArrayList<Ffaviviendaalumbrado>();  
Istvivcoc = new ArrayList<Ffaviviendacocina>();  
Istvivtran = new ArrayList<Ffaviviendatransporte>();  
Istvivi tiempo = new ArrayList<Ffaviviendatiempo>();  
Istrmedioamb = new ArrayList<Ffaviviendariesgosmedio>();  
Istranim = new ArrayList<Ffaviviendariesgosanimal>();  
Istrsanitario = new ArrayList<Ffaviviendariesgossanitario>();  
Istcontentr = new ArrayList<Ffaviviendaentrevista>();
```

```
registrarOpcionesMultiples();  
registrarOpcionesCanton();  
registrarOpcionesEstab();  
registrarOpcionesParr();  
registrarOpcionesDist();  
registrarOpcionesZona();  
registrarOpcionesLoc();  
registrarOpcionesSec();  
registrarOpcionesMan();  
registrarOpcionesDiv();  
registrarOpcionesEdi();  
registrarOpcionesVnum();  
registrarOpcionesVhog();  
registrarOpcionesVocup();  
registrarOpcionesVtip();  
registrarOpcionesVacc();  
registrarOpcionesVtec();  
registrarOpcionesVpis();  
registrarOpcionesVpar();  
registrarOpcionesVest();  
registrarOpcionesVdhog();
```

```

    registrarOpcionesVag();
    registrarOpcionesVagrec();
    registrarOpcionesVagtrat();
    registrarOpcionesVsh();
    registrarOpcionesVshubic();
    registrarOpcionesVsd();
    registrarOpcionesVbas();
    registrarOpcionesValum();
    registrarOpcionesVcoc();
    registrarOpcionesVtrans();
    registrarOpcionesVtiempo();
    registrarOpcionesVrmed();
    registrarOpcionesVranim();
    registrarOpcionesVrsan();
    registrarOpcionesVentrev();

}

```

```

public void buscarPaciente(){
    FacesContext lContexto= FacesContext.getCurrentInstance();
    Ffavivienda_ubicacion buscar = null;

    if (!(cedula.equals(""))){
        try {
            buscar = vguardar.finbyCedula(cedula);
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    if (buscar == null) {

```

```

        IContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_ERROR,"Error","Paciente no
Existe"));

        //return "";
    } else if (buscar != null) {
        IContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_INFO,"Información ","Paciente si
existe en el sistema"));

        //completo=pacienteaut.getPrimernombre()+"
"+pacienteaut.getSegundonombre()+" "+pacienteaut.getApellidoaterno()+"
"+pacienteaut.getApellidomaterno());

        //System.out.println(buscar.getFecha());

        ident=buscar.getIdentificacion();//0
        pnomb=buscar.getNombre_primer();//2
        snomb=buscar.getNombre_segundo();//3
        papell=buscar.getApellido_paterno();//4
        sapell=buscar.getApellido_materno();
        establecimiento=buscar.getEstablecimiento();
        provinc=buscar.getProvincia();
        canton=buscar.getCantones();
        parroquia=buscar.getParroquia();
        distrito=buscar.getDistrito();
        zona=buscar.getZona();
        localidad=buscar.getLocalidad();
        sector=buscar.getSector();
        manzana=buscar.getManzana();
        division=buscar.getDivision();
        edificio=buscar.getEdificio();
        numvivienda=buscar.getViviendanumero();
        numhogar=buscar.getViviendahogar();
        pcalle=buscar.getCalleprincipal();
        scalle=buscar.getCallesecundaria();
        numcasa=buscar.getNumcasa());

```

```
conjhabitac=buscar.getNombrehabitacional();
numlote=buscar.getIdlote();
numdepart=buscar.getIddepartamento();
numpiso=buscar.getIdpiso();
telefono=buscar.getConvencional();
celular=buscar.getCelular();
refvivienda=buscar.getReferenciahabit();
estvivienda=buscar.getViviendaocupacion();
tipvivienda=buscar.getViviendatipos();
accvivienda=buscar.getViviendaacceso();
techovivienda=buscar.getViviendatecho();
pisovivienda=buscar.getViviendapiso();
paredvivienda=buscar.getViviendaparedes();
esttecvivienda=buscar.getEsttechvivienda();
estpisovivienda=buscar.getEstpisovivienda();
estparedvivienda=buscar.getEstparedvivienda();
estocupvivienda=buscar.getEstocupvivienda();
cantcuartos=buscar.getCantcuartos();
cantdormitorio=buscar.getCantdormitorios();
procagua=buscar.getViviendaagua();
recibagua=buscar.getViviendaagrec();
tratagua=buscar.getViviendaagtrat();
tipservhig=buscar.getViviendash();
ubicshig=buscar.getViviendashubic();
servducha=buscar.getViviendaservducha();
elimbas=buscar.getViviendabasura();
tipalumb=buscar.getViviendaalumbrado();
tipcocina=buscar.getViviendacocina();
tiptransp=buscar.getViviendatransporte();
tiempdespl=buscar.getViviendatiempo();
fechausu=buscar.getFecusu();
fechamod=buscar.getFecmod();
codusu=buscar.getCodusu();
codmod=buscar.getCodmod();
```

```

        estado=buscar.getEstado();

rmedioambvivienda=buscar.getRmedioambvivienda();
        ranimal=buscar.getRanimal();
        rsanitario=buscar.getRsanitario();
        contentr=buscar.getContentr();
        nombenc=buscar.getNombenc();
        codenc=buscar.getCodenc();
        fecha=buscar.getFecha();
        numvis=buscar.getNumvis();
    }

    cedula="";

    habpaciente=true;

}

public void registrarOpcionesMultiples(){
    Long idpadre = 1L;
    try {
        IstLocalizacion= localizacionDao.listarporPadre(idpadre);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesCanton(){
    //String idtipo = 1L;
    try {
        IstLocalizacionCanton=
cantonesDao.listarporPadre();//(idtipo);
    } catch (Exception e) {

```

```

        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesEstab(){
    //Long idsector = 11L;
    try {
        IstLocalizacionEstab=
establecimientosaludDao.listarporPadre();
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesParr(){
    //Long tipo = 1L;
    try {
        Istparroquia= parrDao.listarporPadre();
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesDist(){
    //String idtipo = 1L;
    try {
        Istdistrito= distDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

}

public void registrarOpcionesZona(){
    //String idtipo = 1L;
    try {
        lstzona= zonaDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesLoc(){
    //String idtipo = 1L;
    try {
        lstlocal= locDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesSec(){
    //String idtipo = 1L;
    try {
        lstsector= secDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesMan(){
    //String idtipo = 1L;

```

```

    try {
        lstmanzana= manDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesDiv(){
    //String idtipo = 1L;
    try {
        lstdivision= divDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesEdi(){
    //String idtipo = 1L;
    try {
        lstedificio= ediDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVnum(){
    //String idtipo = 1L;
    try {
        lstvivnumero= vnumDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block

```

```

        e.printStackTrace();
    }
}

public void registrarOpcionesVhog(){
    //String idtipo = 1L;
    try {
        Istvivhogar= vhogDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVocup(){
    //String idtipo = 1L;
    try {
        Istvivocup= vocupDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVtip(){
    //String idtipo = 1L;
    try {
        Istvivtip= vtipDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}

```

```

public void registrarOpcionesVacc(){
    //String idtipo = 1L;
    try {
        lstvivacc= vaccDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVtec(){
    //String idtipo = 1L;
    try {
        lstvivtec= vtecDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVpis(){
    //String idtipo = 1L;
    try {
        lstvivpis= vpisDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVpar(){
    //String idtipo = 1L;
    try {
        lstvivpar= vparDao.listarporPadre();//(idtipo);
    }
}

```

```

    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVest(){
    //String idtipo = 1L;
    try {
        lstvinvest= vestDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVdhog(){
    //String idtipo = 1L;
    try {
        lstvvdhogar= vdhogDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVag(){
    //String idtipo = 1L;
    try {
        lstvivagua= vagDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

}

public void registrarOpcionesVagrec(){
    //String idtipo = 1L;
    try {
        lstvivagrec= vagrecDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVagtrat(){
    //String idtipo = 1L;
    try {
        lstvivagtrat= vagtratDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVsh(){
    //String idtipo = 1L;
    try {
        lstvivsh= vshDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVshubic(){
    //String idtipo = 1L;

```

```

    try {
        lstvivshubic= vshubicDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVsd(){
    //String idtipo = 1L;
    try {
        lstvivsd= vsdDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVbas(){
    //String idtipo = 1L;
    try {
        lstvivbas= vbasDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesValum(){
    //String idtipo = 1L;
    try {
        lstvivalum= valumDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block

```

```

        e.printStackTrace();
    }
}

public void registrarOpcionesVcoc(){
    //String idtipo = 1L;
    try {
        Istvivcoc= vcocDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVtrans(){
    //String idtipo = 1L;
    try {
        Istvivtran= vtransDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesVtiempo(){
    //String idtipo = 1L;
    try {
        Istvivtiempo= vtiempoDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}

```

```

public void registrarOpcionesVrmed(){
    //String idtipo = 1L;
    try {
        lstrmedioamb= vrmedDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVrsan(){
    //String idtipo = 1L;
    try {
        lstrsanitario= vrsanDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVranim(){
    //String idtipo = 1L;
    try {
        lstranim= vraniDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

public void registrarOpcionesVentrev(){
    //String idtipo = 1L;
    try {
        lstrcontentr= ventrevDao.listarporPadre();//(idtipo);
    }
}

```

```

    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void actualizar() {
    try {
        //personaDao.merge(this.nuevaPersona);
        //pacienteDao.merge(this.nPaciente);
        //sectorDao.merge(this.nuevoSector);
        //familiaDao.merge(this.nuevaFamilia);
        localizacionDao.merge(local);
        cantonesDao.merge(cantones);
        distDao.merge(dist);
        establecimientosaludDao.merge(estab);
        presentarMensaje("I", "Aplicacion", "Actualizacion exitosa");
    } catch (Exception e) {
        presentarMensaje("E", "Error", "Error en actualizacion:"+e);
    }
}
}

```

```

public String guardar(){

    System.out.println("Guardar()");
    int idubic = 0;
    //Guardar datos en tabla
    try {

        vtabguardar =
vguardar.guardarVivienda_ubicacion(idubic,//1

        ident,//0

```

pnomb,//2

snomb,//3

papell,//4

sapell,//5

establecimiento,//6

provinc,//7

canton,//8

parroquia,//9

distrito,//10

zona,//11

localidad,//12

sector,//13

manzana,//14

division,//15

edificio,//16

numvivienda,//17

numhogar,//18

pcalle,//19

scalle,//20

numcasa,//21

conjhabitac,//22

numlote,//23

numdepart,//24

numpiso,//25

telefono,//26

celular,//27

refvivienda,//28

estvivienda,//29

tipvivienda,//30

accvivienda,//31

techovivienda,//32

pisovivienda,//33

paredvivienda,//34

esttecvivienda,//35

estpisovivienda,//36

estparedvivienda,//37

estocupvivienda,//38

cantcuartos,//39

cantdormitorio,//40

procagua,//41

recibagua,//42

tratagua,//43

tipservhig,//44

ubicservhig,//45

servducha,//46

elimbas,//47

tipalumb,//48

tipcocina,//49

tiptransp,//50

tiempdespl,//51

fechausu,//52

fechamod,//53

codusu,//54

codmod,//55

estado,//56

rmedioambvivienda,//56

ranimal,//57

rsanitario,//58

contentr,//59

nombenc,//60

codenc,//61

fecha,//62

numvis//63

);

//cedula = "";

ident= "";

pnomb= "";

snomb= "";

papell= "";

sapell= "";

establecimiento= "";
provinc= "";
canton= "";
parroquia= "";
distrito= "";
zona= "";
localidad= "";
sector= "";
manzana= "";
division= "";
edificio= "";
numvivienda= "";
numhogar= "";
pcalle= "";
scalle= "";
numcasa= "";
conjhabitac= "";
numlote= "";
numdepart= "";
numpiso= "";
telefono= "";
celular= "";
refvivienda= "";
estvivienda= "";
tipvivienda= "";
accvivienda= "";
techovivienda= "";
pisovivienda= "";
paredvivienda= "";
esttecvivienda= "";
estpisovivienda= "";
estparedvivienda= "";
estocupvivienda= "";
cantcuartos= "";

```
cantdormitorio= "";
procagua= "";
recibagua= "";
tratagua= "";
tipservhig= "";
ubicservhig= "";
servducha= "";
elimbas= "";
tipalumb= "";
tipcocina= "";
tiptransp= "";
tiempdespl= "";
fechausu = null;
fechamod = null;
codusu= "";
codmod= "";
estado= "";
rmedioambvivienda= "";
ranimal= "";
rsanitario= "";
contentr= "";
nombenc= "";
codenc= "";
fecha = null;
numvis= "";

        System.out.println("\ntabguardar(") );

    } catch (Exception e1) {e1.printStackTrace();}

    return "";//redireccionar;
}
```

```

        public void presentarMensaje(String severity, String title, String
aplicacion) {
            FacesContext context = FacesContext.getCurrentInstance();
            if(severity.equals("I"))
                context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_INFO,title, aplicacion));
            if(severity.equals("W"))
                context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_WARN,title, aplicacion));
            if(severity.equals("E"))
                context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_ERROR,title, aplicacion));
            if(severity.equals("F"))
                context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_FATAL,title, aplicacion));

        }

        public void refresh(){

                FacesContext context = FacesContext.getCurrentInstance();
                Application application = context.getApplication();
                ViewHandler viewHandler = application.getViewHandler();
                UIViewRoot viewRoot = viewHandler.createView(context,
context.getViewRoot().getViewId());
                context.setViewRoot(viewRoot);
                context.renderResponse();
        }

        public void limpiar()
        {
                //Método que limpia el formulario en pantalla

                habpaciente=false;

```

```
//Limpia formulario
cedula = "";
ident= "";
pnomb= "";
snomb= "";
papell= "";
sapell= "";
establecimiento= "";
provinc= "";
canton= "";
parroquia= "";
distrito= "";
zona= "";
localidad= "";
sector= "";
manzana= "";
division= "";
edificio= "";
numvivienda= "";
numhogar= "";
pcalle= "";
scalle= "";
numcasa= "";
conjhabitac= "";
numlote= "";
numdepart= "";
numpiso= "";
telefono= "";
celular= "";
refvivienda= "";
estvivienda= "";
tipvivienda= "";
accvivienda= "";
techovivienda= "";
```

```
    pisovivienda= "";
    paredvivienda= "";
    esttecivivienda= "";
    estpisovivienda= "";
    estparedvivienda= "";
    estocupvivienda= "";
    cantcuartos= "";
    cantdormitorio= "";
    procagua= "";
    recibagua= "";
    tratagua= "";
    tipservhig= "";
    ubicservhig= "";
    servducha= "";
    elimbas= "";
    tipalumb= "";
    tipcocina= "";
    tiptransp= "";
    tiempdespl= "";
    fechausu = null;
    fechamod = null;
    codusu= "";
    codmod= "";
    estado= "";
    rmedioambvivienda= "";
    ranimal= "";
    rsanitario= "";
    contentr= "";
nombenc= "";
    codenc= "";
    fecha = null;
    numvis= "";
}
```

```
public String getIdent() {
    return ident;
}

public void setIdent(String ident) {
    this.ident = ident;
}

public String getLocalidad() {
    return localidad;
}

public void setLocalidad(String localidad) {
    this.localidad = localidad;
}

public String getDivision() {
    return division;
}

public void setDivision(String division) {
    this.division = division;
}

public String getEdificio() {
    return edificio;
}

public void setEdificio(String edificio) {
    this.edificio = edificio;
}

public String getConjhabitac() {
```

```
        return conjhabitac;
    }

    public void setConjhabitac(String conjhabitac) {
        this.conjhabitac = conjhabitac;
    }

    public String getCelular() {
        return celular;
    }

    public void setCelular(String celular) {
        this.celular = celular;
    }

    public String getEstvivienda() {
        return estvivienda;
    }

    public void setEstvivienda(String estvivienda) {
        this.estvivienda = estvivienda;
    }

    public String getAccvivienda() {
        return accvivienda;
    }

    public void setAccvivienda(String accvivienda) {
        this.accvivienda = accvivienda;
    }

    public String getEsttecvivienda() {
        return esttecvivienda;
    }
}
```

```
public void setEsttecvivienda(String esttecvivienda) {
    this.esttecvivienda = esttecvivienda;
}

public String getEstpisovivienda() {
    return estpisovivienda;
}

public void setEstpisovivienda(String estpisovivienda) {
    this.estpisovivienda = estpisovivienda;
}

public String getEstparedvivienda() {
    return estparedvivienda;
}

public void setEstparedvivienda(String estparedvivienda) {
    this.estparedvivienda = estparedvivienda;
}

public String getEstocupvivienda() {
    return estocupvivienda;
}

public void setEstocupvivienda(String estocupvivienda) {
    this.estocupvivienda = estocupvivienda;
}

public String getCantcuartos() {
    return cantcuartos;
}

public void setCantcuartos(String cantcuartos) {
```

```
        this.cantcuartos = cantcuartos;
    }

    public String getCantdormitorio() {
        return cantdormitorio;
    }

    public void setCantdormitorio(String cantdormitorio) {
        this.cantdormitorio = cantdormitorio;
    }

    public String getElimbas() {
        return elimbas;
    }

    public void setElimbas(String elimbas) {
        this.elimbas = elimbas;
    }

    public Date getFechausu() {
        return fechausu;
    }

    public void setFechausu(Date fechausu) {
        this.fechausu = fechausu;
    }

    public Date getFechamod() {
        return fechamod;
    }

    public void setFechamod(Date fechamod) {
        this.fechamod = fechamod;
    }
}
```

```
public String getCodusu() {
    return codusu;
}

public void setCodusu(String codusu) {
    this.codusu = codusu;
}

public String getCodmod() {
    return codmod;
}

public void setCodmod(String codmod) {
    this.codmod = codmod;
}

public String getEstado() {
    return estado;
}

public void setEstado(String estado) {
    this.estado = estado;
}

public String getPnomb() {
    return pnomb;
}

public void setPnomb(String pnomb) {
    this.pnomb = pnomb;
}

public String getSnomb() {
```

```
        return snomb;
    }

    public void setSnomb(String snomb) {
        this.snomb = snomb;
    }

    public String getPapell() {
        return papell;
    }

    public void setPapell(String papell) {
        this.papell = papell;
    }

    public String getSapell() {
        return sapell;
    }

    public void setSapell(String sapell) {
        this.sapell = sapell;
    }

    public String getSector() {
        return sector;
    }

    public void setSector(String sector) {
        this.sector = sector;
    }

    public String getManzana() {
        return manzana;
    }
}
```

```
public void setManzana(String manzana) {
    this.manzana = manzana;
}

public String getNumvivienda() {
    return numvivienda;
}

public void setNumvivienda(String numvivienda) {
    this.numvivienda = numvivienda;
}

public String getNumhogar() {
    return numhogar;
}

public void setNumhogar(String numhogar) {
    this.numhogar = numhogar;
}

public String getPcalle() {
    return pcalle;
}

public void setPcalle(String pcalle) {
    this.pcalle = pcalle;
}

public String getScalle() {
    return scalle;
}

public void setScalle(String scalle) {
```

```
        this.scalle = scalle;
    }

    public String getNumcasa() {
        return numcasa;
    }

    public void setNumcasa(String numcasa) {
        this.numcasa = numcasa;
    }

    public String getNumlote() {
        return numlote;
    }

    public void setNumlote(String numlote) {
        this.numlote = numlote;
    }

    public String getNumdepart() {
        return numdepart;
    }

    public void setNumdepart(String numdepart) {
        this.numdepart = numdepart;
    }

    public String getNumpiso() {
        return numpiso;
    }

    public void setNumpiso(String numpiso) {
        this.numpiso = numpiso;
    }
}
```

```
public String getTelefono() {
    return telefono;
}

public void setTelefono(String telefono) {
    this.telefono = telefono;
}

public String getRefvivienda() {
    return refvivienda;
}

public void setRefvivienda(String refvivienda) {
    this.refvivienda = refvivienda;
}

public String getTipvivienda() {
    return tipvivienda;
}

public void setTipvivienda(String tipvivienda) {
    this.tipvivienda = tipvivienda;
}

public String getTechovivienda() {
    return techovivienda;
}

public void setTechovivienda(String techovivienda) {
    this.techovivienda = techovivienda;
}

public String getPisovivienda() {
```

```
        return pisovivienda;
    }

    public void setPisovivienda(String pisovivienda) {
        this.pisovivienda = pisovivienda;
    }

    public String getParedvivienda() {
        return paredvivienda;
    }

    public void setParedvivienda(String paredvivienda) {
        this.paredvivienda = paredvivienda;
    }

    public String getProcagua() {
        return procagua;
    }

    public void setProcagua(String procagua) {
        this.procagua = procagua;
    }

    public String getRecibagua() {
        return recibagua;
    }

    public void setRecibagua(String recibagua) {
        this.recibagua = recibagua;
    }

    public String getTratagua() {
        return tratagua;
    }
}
```

```
public void setTratagua(String tratagua) {
    this.tratagua = tratagua;
}

public String getTipservhig() {
    return tipservhig;
}

public void setTipservhig(String tipservhig) {
    this.tipservhig = tipservhig;
}

public String getUbicservhig() {
    return ubicservhig;
}

public void setUbicservhig(String ubicservhig) {
    this.ubicservhig = ubicservhig;
}

public String getServducha() {
    return servducha;
}

public void setServducha(String servducha) {
    this.servducha = servducha;
}

public String getTupalumb() {
    return tupalumb;
}

public void setTupalumb(String tupalumb) {
```

```

        this.tipalumb = tipalumb;
    }

    public String getTipcocina() {
        return tipcocina;
    }

    public void setTipcocina(String tipcocina) {
        this.tipcocina = tipcocina;
    }

    public String getTiptransp() {
        return tiptransp;
    }

    public void setTiptransp(String tiptransp) {
        this.tiptransp = tiptransp;
    }

    public String getTiempdespl() {
        return tiempdespl;
    }

    public void setTiempdespl(String tiempdespl) {
        this.tiempdespl = tiempdespl;
    }

    //////////////////////////////////////
    /*public Parpersona getNuevaPersona() {
        return nuevaPersona;
    }

```

```

public void setNuevaPersona(Parpersona nuevaPersona) {
    this.nuevaPersona = nuevaPersona;
}

public Parfamilia getNuevaFamilia() {
    return nuevaFamilia;
}

public void setNuevaFamilia(Parfamilia nuevaFamilia) {
    this.nuevaFamilia = nuevaFamilia;
}

public Atmsectore getNuevoSector() {
    return nuevoSector;
}

public void setNuevoSector(Atmsectore nuevoSector) {
    this.nuevoSector = nuevoSector;
}

public Parpaciente getnPaciente() {
    return nPaciente;
}

public void setnPaciente(Parpaciente nPaciente) {
    this.nPaciente = nPaciente;
}*/

public HcfLocalizacion getLocal() {
    return local;
}

public void setLocal(HcfLocalizacion local) {
    this.local = local;
}

```

```

    }

    public Map<String, String> getEstablec() {
        return establec;
    }

    public void setEstablec(Map<String, String> establec) {
        this.establec = establec;
    }

    public String getEstablecimiento() {
        return establecimiento;
    }

    public void setEstablecimiento(String establecimiento) {
        this.establecimiento = establecimiento;
    }

    public List<HcfLocalizacion> getLstLocalizacion() {
        return lstLocalizacion;
    }

    public void setLstLocalizacion(List<HcfLocalizacion> lstLocalizacion) {
        this.lstLocalizacion = lstLocalizacion;
    }

    public List<ProEstablecimientoSalud> getLstLocalizacionEstab() {
        return lstLocalizacionEstab;
    }

    public void setLstLocalizacionEstab(List<ProEstablecimientoSalud>
lstLocalizacionEstab) {
        this.lstLocalizacionEstab = lstLocalizacionEstab;
    }
}

```

```

public String getProvinc() {
    return provinc;
}

public void setProvinc(String provinc) {
    this.provinc = provinc;
}

public List<Ffacantones> getLstLocalizacionCanton() {
    return lstLocalizacionCanton;
}

public void setLstLocalizacionCanton(List<Ffacantones>
lstLocalizacionCanton) {
    this.lstLocalizacionCanton = lstLocalizacionCanton;
}

public String getCanton() {
    return canton;
}

public void setCanton(String canton) {
    this.canton = canton;
}

public Ffacantones getCantones() {
    return cantones;
}

public void setCantones(Ffacantones cantones) {
    this.cantones = cantones;
}

```

```

public String getParroquia() {
    return parroquia;
}

public void setParroquia(String parroquia) {
    this.parroquia = parroquia;
}

public List<Ffaparroquia> getIstparroquia() {
    return Istparroquia;
}

public void setIstparroquia(List<Ffaparroquia> Istparroquia) {
    this.Istparroquia = Istparroquia;
}

public Ffaparroquia getParr() {
    return parr;
}

public void setParr(Ffaparroquia parr) {
    this.parr = parr;
}

public Ffadistrito getDist() {
    return dist;
}

public void setDist(Ffadistrito dist) {
    this.dist = dist;
}

public String getDistrito() {
    return distrito;
}

```

```
}

public void setDistrito(String distrito) {
    this.distrito = distrito;
}

public List<Ffadistrito> getLstdistrito() {
    return lstdistrito;
}

public void setLstdistrito(List<Ffadistrito> lstdistrito) {
    this.lstdistrito = lstdistrito;
}

public Ffazona getZon() {
    return zon;
}

public void setZon(Ffazona zon) {
    this.zon = zon;
}

public String getZona() {
    return zona;
}

public void setZona(String zona) {
    this.zona = zona;
}

public List<Ffasector> getLstsector() {
    return lstsector;
}
```

```
public void setLstsector(List<Ffasector> lstsector) {  
    this.lstsector = lstsector;  
}
```

```
public Ffasector getSex() {  
    return sex;  
}
```

```
public void setSex(Ffasector sex) {  
    this.sex = sex;  
}
```

```
public Ffalocalidad getLoc() {  
    return loc;  
}
```

```
public void setLoc(Ffalocalidad loc) {  
    this.loc = loc;  
}
```

```
public List<Ffalocalidad> getLstlocal() {  
    return lstlocal;  
}
```

```
public void setLstlocal(List<Ffalocalidad> lstlocal) {  
    this.lstlocal = lstlocal;  
}
```

```
public List<Ffazona> getLstzona() {  
    return lstzona;  
}
```

```
public void setLstzona(List<Ffazona> lstzona) {  
    this.lstzona = lstzona;  
}
```

```

}

public Ffamanzana getManz() {
    return manz;
}

public void setManz(Ffamanzana manz) {
    this.manz = manz;
}

public List<Ffamanzana> getLstmanzana() {
    return lstmanzana;
}

public void setLstmanzana(List<Ffamanzana> lstmanzana) {
    this.lstmanzana = lstmanzana;
}

public Ffadivision getDiv() {
    return div;
}

public void setDiv(Ffadivision div) {
    this.div = div;
}

public List<Ffadivision> getLstdivision() {
    return lstdivision;
}

public void setLstdivision(List<Ffadivision> lstdivision) {
    this.lstdivision = lstdivision;
}

```

```

public Ffaedificio getEdi() {
    return edi;
}

public void setEdi(Ffaedificio edi) {
    this.edi = edi;
}

public List<Ffaedificio> getLstedificio() {
    return lstedificio;
}

public void setLstedificio(List<Ffaedificio> lstedificio) {
    this.lstedificio = lstedificio;
}

public Ffaviviendanumero getVnum() {
    return vnum;
}

public void setVnum(Ffaviviendanumero vnum) {
    this.vnum = vnum;
}

public List<Ffaviviendanumero> getLstvivnumero() {
    return lstvivnumero;
}

public void setLstvivnumero(List<Ffaviviendanumero> lstvivnumero) {
    this.lstvivnumero = lstvivnumero;
}

public Ffaviviendahogar getVhog() {
    return vhog;
}

```

```
}
```

```
public void setVhog(Ffaviviendahogar vhog) {  
    this.vhog = vhog;  
}
```

```
}
```

```
public List<Ffaviviendahogar> getLstvivhogar() {  
    return lstvivhogar;  
}
```

```
}
```

```
public void setLstvivhogar(List<Ffaviviendahogar> lstvivhogar) {  
    this.lstvivhogar = lstvivhogar;  
}
```

```
}
```

```
public Ffaviviendaocupacion getVestocup() {  
    return vestocup;  
}
```

```
}
```

```
public void setVestocup(Ffaviviendaocupacion vestocup) {  
    this.vestocup = vestocup;  
}
```

```
}
```

```
public List<Ffaviviendaocupacion> getLstvivocup() {  
    return lstvivocup;  
}
```

```
}
```

```
public void setLstvivocup(List<Ffaviviendaocupacion> lstvivocup) {  
    this.lstvivocup = lstvivocup;  
}
```

```
}
```

```
public Ffaviviendatipos getVtipos() {  
    return vtipos;  
}
```

```
}
```

```

public void setVtipos(Ffaviviendatipos vtipos) {
    this.vtipos = vtipos;
}

public List<Ffaviviendatipos> getLstvivtip() {
    return lstvivtip;
}

public void setLstvivtip(List<Ffaviviendatipos> lstvivtip) {
    this.lstvivtip = lstvivtip;
}

public Ffaviviendaacceso getVaccesso() {
    return vaccesso;
}

public void setVaccesso(Ffaviviendaacceso vaccesso) {
    this.vaccesso = vaccesso;
}

public List<Ffaviviendaacceso> getLstvivacc() {
    return lstvivacc;
}

public void setLstvivacc(List<Ffaviviendaacceso> lstvivacc) {
    this.lstvivacc = lstvivacc;
}

public Ffaviviendatecho getVtecho() {
    return vtecho;
}

public void setVtecho(Ffaviviendatecho vtecho) {
    this.vtecho = vtecho;
}

```

```

}

public List<Ffaviviendatecho> getLstvivtec() {
    return lstvivtec;
}

public void setLstvivtec(List<Ffaviviendatecho> lstvivtec) {
    this.lstvivtec = lstvivtec;
}

public Ffaviviendapiso getVpiso() {
    return vpiso;
}

public void setVpiso(Ffaviviendapiso vpiso) {
    this.vpiso = vpiso;
}

public List<Ffaviviendapiso> getLstvivpis() {
    return lstvivpis;
}

public void setLstvivpis(List<Ffaviviendapiso> lstvivpis) {
    this.lstvivpis = lstvivpis;
}

public Ffaviviendaparedes getVparedes() {
    return vparedes;
}

public void setVparedes(Ffaviviendaparedes vparedes) {
    this.vparedes = vparedes;
}

```

```

public List<Ffaviviendaparedes> getLstvivpar() {
    return lstvivpar;
}

public void setLstvivpar(List<Ffaviviendaparedes> lstvivpar) {
    this.lstvivpar = lstvivpar;
}

public Ffaviviendaestado getVestado() {
    return vestado;
}

public void setVestado(Ffaviviendaestado vestado) {
    this.vestado = vestado;
}

public List<Ffaviviendaestado> getLstvivest() {
    return lstvivest;
}

public void setLstvivest(List<Ffaviviendaestado> lstvivest) {
    this.lstvivest = lstvivest;
}

public Ffaviviendadhogar getVdhog() {
    return vdhog;
}

public void setVdhog(Ffaviviendadhogar vdhog) {
    this.vdhog = vdhog;
}

public List<Ffaviviendadhogar> getLstvivdhogar() {
    return lstvivdhogar;
}

```

```

}

public void setLstvivdhogar(List<Ffaviviendadhogar> lstvivdhogar) {
    this.lstvivdhogar = lstvivdhogar;
}

public Ffaviviendaagua getVagua() {
    return vagua;
}

public void setVagua(Ffaviviendaagua vagua) {
    this.vagua = vagua;
}

public List<Ffaviviendaagua> getLstvivagua() {
    return lstvivagua;
}

public void setLstvivagua(List<Ffaviviendaagua> lstvivagua) {
    this.lstvivagua = lstvivagua;
}

public Ffaviviendaagrec getVagrec() {
    return vagrec;
}

public void setVagrec(Ffaviviendaagrec vagrec) {
    this.vagrec = vagrec;
}

public List<Ffaviviendaagrec> getLstvivagrec() {
    return lstvivagrec;
}

```

```
public void setLstvivagrec(List<Ffaviviendaagrec> lstvivagrec) {  
    this.lstvivagrec = lstvivagrec;  
}
```

```
public Ffaviviendaagrec getVagtrat() {  
    return vagtrat;  
}
```

```
public void setVagtrat(Ffaviviendaagrec vagtrat) {  
    this.vagtrat = vagtrat;  
}
```

```
public List<Ffaviviendaagtrat> getLstvivagtrat() {  
    return lstvivagtrat;  
}
```

```
public void setLstvivagtrat(List<Ffaviviendaagtrat> lstvivagtrat) {  
    this.lstvivagtrat = lstvivagtrat;  
}
```

```
public Ffaviviendash getVsh() {  
    return vsh;  
}
```

```
public void setVsh(Ffaviviendash vsh) {  
    this.vsh = vsh;  
}
```

```
public List<Ffaviviendash> getLstvivsh() {  
    return lstvivsh;  
}
```

```
public void setLstvivsh(List<Ffaviviendash> lstvivsh) {  
    this.lstvivsh = lstvivsh;  
}
```

```

}

public Ffaviviendashubic getVshubic() {
    return vshubic;
}

public void setVshubic(Ffaviviendashubic vshubic) {
    this.vshubic = vshubic;
}

public List<Ffaviviendashubic> getLstvivshubic() {
    return lstvivshubic;
}

public void setLstvivshubic(List<Ffaviviendashubic> lstvivshubic) {
    this.lstvivshubic = lstvivshubic;
}

public Ffaviviendaservducha getVserduc() {
    return vserduc;
}

public void setVserduc(Ffaviviendaservducha vserduc) {
    this.vserduc = vserduc;
}

public List<Ffaviviendaservducha> getLstvivsd() {
    return lstvivsd;
}

public void setLstvivsd(List<Ffaviviendaservducha> lstvivsd) {
    this.lstvivsd = lstvivsd;
}

```

```

public Ffaviviendabasura getVbasura() {
    return vbasura;
}

public void setVbasura(Ffaviviendabasura vbasura) {
    this.vbasura = vbasura;
}

public List<Ffaviviendabasura> getLstvivbas() {
    return lstvivbas;
}

public void setLstvivbas(List<Ffaviviendabasura> lstvivbas) {
    this.lstvivbas = lstvivbas;
}

public Ffaviviendaalumbrado getValumbrado() {
    return valumbrado;
}

public void setValumbrado(Ffaviviendaalumbrado valumbrado) {
    this.valumbrado = valumbrado;
}

public List<Ffaviviendaalumbrado> getLstvivalum() {
    return lstvivalum;
}

public void setLstvivalum(List<Ffaviviendaalumbrado> lstvivalum) {
    this.lstvivalum = lstvivalum;
}

public Ffaviviendacocina getVcocina() {
    return vcocina;
}

```

```

}

public void setVcocina(Ffaviviendacocina vcocina) {
    this.vcocina = vcocina;
}

public List<Ffaviviendacocina> getLstvivcoc() {
    return lstvivcoc;
}

public void setLstvivcoc(List<Ffaviviendacocina> lstvivcoc) {
    this.lstvivcoc = lstvivcoc;
}

public Ffaviviendatransporte getVtrans() {
    return vtrans;
}

public void setVtrans(Ffaviviendatransporte vtrans) {
    this.vtrans = vtrans;
}

public List<Ffaviviendatransporte> getLstvivtran() {
    return lstvivtran;
}

public void setLstvivtran(List<Ffaviviendatransporte> lstvivtran) {
    this.lstvivtran = lstvivtran;
}

public Ffaviviendatiempo getVtiempo() {
    return vtiempo;
}

```

```
public void setVtiempo(Ffaviviendatiempo vtiempo) {
    this.vtiempo = vtiempo;
}

public List<Ffaviviendatiempo> getLstvivitiempo() {
    return lstvivitiempo;
}

public void setLstvivitiempo(List<Ffaviviendatiempo> lstvivitiempo) {
    this.lstvivitiempo = lstvivitiempo;
}

public String getCedula() {
    return cedula;
}

public void setCedula(String cedula) {
    this.cedula = cedula;
}

public String getRmedioambvivienda() {
    return rmedioambvivienda;
}

public void setRmedioambvivienda(String rmedioambvivienda) {
    this.rmedioambvivienda = rmedioambvivienda;
}

public String getRanimal() {
    return ranimal;
}

public void setRanimal(String ranimal) {
```

```
        this.ranimal = ranimal;
    }

    public String getRsanitario() {
        return rsanitario;
    }

    public void setRsanitario(String rsanitario) {
        this.rsanitario = rsanitario;
    }

    public Date getFecha() {
        return fecha;
    }

    public void setFecha(Date fecha) {
        this.fecha = fecha;
    }

    public String getNumvis() {
        return numvis;
    }

    public void setNumvis(String numvis) {
        this.numvis = numvis;
    }

    public String getContentr() {
        return contentr;
    }

    public void setContentr(String contentr) {
        this.contentr = contentr;
    }
}
```

```

public String getNombenc() {
    return nombenc;
}

public void setNombenc(String nombenc) {
    this.nombenc = nombenc;
}

public String getCodenc() {
    return codenc;
}

public void setCodenc(String codenc) {
    this.codenc = codenc;
}

public List<Ffaviviendariesgosmedio> getLstrmedioamb() {
    return lstrmedioamb;
}

public void setLstrmedioamb(List<Ffaviviendariesgosmedio>
lstrmedioamb) {
    this.lstrmedioamb = lstrmedioamb;
}

public List<Ffaviviendariesgosanimal> getLstranim() {
    return lstranim;
}

public void setLstranim(List<Ffaviviendariesgosanimal> lstranim) {
    this.lstranim = lstranim;
}

```

```

        public List<Ffaviviendariesgossanitario> getLstrsanitario() {
            return lstrsanitario;
        }

        public void setLstrsanitario(List<Ffaviviendariesgossanitario>
lstrsanitario) {
            this.lstrsanitario = lstrsanitario;
        }

        public List<Ffaviviendaentrevista> getLstcontentr() {
            return lstcontentr;
        }

        public void setLstcontentr(List<Ffaviviendaentrevista> lstcontentr) {
            this.lstcontentr = lstcontentr;
        }

        public boolean isHabpaciente() {
            return habpaciente;
        }

        public void setHabpaciente(boolean habpaciente) {
            this.habpaciente = habpaciente;
        }
    }

```

Controlador Mortalidad Familiar

```

package edu.cisc.controlador;

import java.io.Serializable;
import java.io.IOException;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;

```

```
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Date;
import java.util.GregorianCalendar;
import java.util.LinkedHashMap;

import javax.annotation.PostConstruct;
import javax.faces.application.Application;
import javax.faces.application.FacesMessage;
import javax.faces.application.ViewHandler;
import javax.faces.bean.SessionScoped;
import javax.faces.bean.ViewScoped;
import javax.faces.component.UIViewRoot;
import javax.faces.context.FacesContext;

import javax.persistence.NoResultException;
import javax.faces.context.ExternalContext;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Qualifier;
import org.springframework.stereotype.Component;

import edu.cisc.dao.FfatiposexosDao;
import edu.cisc.dao.FfamortalidaddocumentacionDao;
import edu.cisc.dao.FfamortalidadparentescoDao;
import edu.cisc.dao.FfamortalidadclaseDao;
import edu.cisc.dao.Ffavivienda_mortalidadDao;

import edu.cisc.modelo.Ffatiposexos;
import edu.cisc.modelo.Ffamortalidaddocumentacion;
import edu.cisc.modelo.Ffamortalidadparentesco;
import edu.cisc.modelo.Ffamortalidadclase;
import edu.cisc.modelo.Ffavivienda_mortalidad;
```

```

@Component(value="mortalidadMB")
@SessionScoped
public class DatosMortalidadMB implements Serializable{

    private static final long serialVersionUID = 1L;

    private Ffavivienda_mortalidad vtabguardar;

    //mortalidad
    private String identif;//1
    private String nombre;//2
    private String apellido;//3
    private String sexo;//4
    private String edad;//5
    private String document;//6
    private String parent;//7
    private Date fecha;//8
    private String clase;//9
    private String causa;//10
    private String observacion;//11
    //para seleccionar la cedula
    private String cedula;
    //para validar que se visualice los botones
    private boolean habpaciente=false;

    private List<Ffatiposexos> Istsexo;
    private List<Ffamortalidaddocumentacion> Istdocum;
    private List<Ffamortalidadparentesco> Istparen;
    private List<Ffamortalidadclase> Istclase;

    @Autowired
    private FfatiposexosDao sexoDao;
    @Autowired

```

```

private FfamortalidaddocumentacionDao documDao;
@Autowired
private FfamortalidadparentescoDao parenDao;
@Autowired
private FfamortalidadclaseDao claseDao;

@Autowired
private Ffavivienda_mortalidadDao vguardar;

@PostConstruct
public void init() {
    //refresh();
    lstsexo = new ArrayList<Ffatiposexos>();
    lstdocum = new ArrayList<Ffamortalidaddocumentacion>();
    lstparen = new ArrayList<Ffamortalidadparentesco>();
    lstclase = new ArrayList<Ffamortalidadclase>();

    registrarOpcionesSexo();
    registrarOpcionesDocumentos();
    registrarOpcionesParentesco();
    registrarOpcionesClase();
}

public void buscarPaciente(){
    FacesContext lContexto= FacesContext.getCurrentInstance();
    Ffavivienda_mortalidad buscar = null;

    if (!(cedula.equals(""))){
        try {
            System.out.println(cedula);
            buscar = vguardar.finbyCedula(cedula);
        } catch (Exception e) {
            // TODO Auto-generated catch block

```

```

        e.printStackTrace();
    }
}

System.out.println(buscar);
if (buscar == null) {
    IContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_ERROR,"Error","Paciente no
Existe"));

    //return "";
} else if (buscar != null) {
    IContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_INFO,"Información ","Paciente si
existe en el sistema"));

    //completo=pacienteaut.getPrimernombre()+
"+pacienteaut.getSegundonombre()+" "+pacienteaut.getApellidopaterno()+"
"+pacienteaut.getApellidomaterno());

    System.out.println(buscar.getFecreg());

    nombre=buscar.getNombre_completo();
    apellido=buscar.getApellido_completo();//3
    sexo=buscar.getSexo();//4
    edad=buscar.getEdad();//5
    document=buscar.getDocumento();//6
    identif=buscar.getIdentificacion();//7
    parent=buscar.getPariente();//8
    fecha=buscar.getFecreg();//9
    clase=buscar.getClase();//10
    causa=buscar.getCausa();//11
    observacion=buscar.getObservacion();//12
}

cedula=" ";
habpaciente=true;

```

```

}

public void registrarOpcionesSexo(){
    //String idtipo = 1L;
    try {
        lstsexo= sexoDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesDocumentos(){
    //String idtipo = 1L;
    try {
        lstdocum= documDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesParentesco(){
    //String idtipo = 1L;
    try {
        lstparen= parenDao.listarporPadre();//(idtipo);
    } catch (Exception e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void registrarOpcionesClase(){
    //String idtipo = 1L;

```

```

        try {
            lstclase= claseDao.listarporPadre();//(idtipo);
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    public void actualizar() {
        try {
            //personaDao.merge(this.nuevaPersona);
            //establecimientosaludDao.merge(estab);
            presentarMensaje("I", "Aplicacion", "Actualizacion exitosa");
        } catch (Exception e) {
            presentarMensaje("E", "Error", "Error en actualizacion:"+e);
        }
    }

    public String guardar(){

        System.out.println("Guardar()");
        int idmort = 0;
        //Guardar datos en tabla intervencion
        try {
            System.out.println(idmort);
            System.out.println(nombre);
            System.out.println(apellido);
            System.out.println(sexo);
            System.out.println(edad);
            System.out.println(document);
            System.out.println(identif);
            System.out.println(parent);
            System.out.println(fecha);
            System.out.println(clase);
        }
    }

```

```
System.out.println(causa);
System.out.println(observacion);
vtabguardar =
vguardar.guardarVivienda_mortalidad(idmort,//1
```

```
nombre,//2
```

```
apellido,//3
```

```
sexo,//4
```

```
edad,//5
```

```
document,//6
```

```
identif,//7
```

```
parent,//8
```

```
fecha,//9
```

```
clase,//10
```

```
causa,//11
```

```
observacion//12
```

```
);
```

```
nombre= "";//2
```

```
apellido= "";//3
```

```
sexo= "";//4
```

```
edad= "";//5
```

```
document= "";//6
```

```

        identif= "";//7
        parent= "";//8
        fecha= null;//9
        clase= "";//10
        causa= "";//11
        observacion= "";//12

        System.out.println("vtabguardar(" );

    } catch (Exception e1) {e1.printStackTrace();}

    return "";//redireccionar;
}

//select
public Ffavivienda_mortalidad getVtabguardar() {
    return vtabguardar;
}

public void setVtabguardar(Ffavivienda_mortalidad vtabguardar) {
    this.vtabguardar = vtabguardar;
}

//insert
public String getIdentif() {
    return identif;
}

public void setIdentif(String identif) {
    this.identif = identif;
}

public String getNombre() {
    return nombre;
}

```

```
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

public String getApellido() {
    return apellido;
}

public void setApellido(String apellido) {
    this.apellido = apellido;
}

public String getSexo() {
    return sexo;
}

public void setSexo(String sexo) {
    this.sexo = sexo;
}

public String getEdad() {
    return edad;
}

public void setEdad(String edad) {
    this.edad = edad;
}

public String getDocument() {
    return document;
}
```

```
public void setDocument(String document) {  
    this.document = document;  
}
```

```
public String getParent() {  
    return parent;  
}
```

```
public void setParent(String parent) {  
    this.parent = parent;  
}
```

```
public Date getFecha() {  
    return fecha;  
}
```

```
public void setFecha(Date fecha) {  
    this.fecha = fecha;  
}
```

```
public String getClase() {  
    return clase;  
}
```

```
public void setClase(String clase) {  
    this.clase = clase;  
}
```

```
public String getCausa() {  
    return causa;  
}
```

```
public void setCausa(String causa) {  
    this.causa = causa;  
}
```

```

    }

    public String getObservacion() {
        return observacion;
    }

    public void setObservacion(String observacion) {
        this.observacion = observacion;
    }

    public String getCedula() {
        return cedula;
    }

    public void setCedula(String cedula) {
        this.cedula = cedula;
    }

    public void presentarMensaje(String severity, String title, String
aplicacion) {
        FacesContext context = FacesContext.getCurrentInstance();
        if(severity.equals("I"))
            context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_INFO,title, aplicacion));
        if(severity.equals("W"))
            context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_WARN,title, aplicacion));
        if(severity.equals("E"))
            context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_ERROR,title, aplicacion));
        if(severity.equals("F"))
            context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_FATAL,title, aplicacion));
    }

```

```
}
```

```
public void refresh(){  
  
    FacesContext context = FacesContext.getCurrentInstance();  
    Application application = context.getApplication();  
    ViewHandler viewHandler = application.getViewHandler();  
    UIViewRoot viewRoot = viewHandler.createView(context,  
context.getViewRoot().getViewId());  
    context.setViewRoot(viewRoot);  
    context.renderResponse();  
  
}
```

```
public void limpiar()  
{  
  
    //Método que limpia el formulario en pantalla  
    habpaciente=false;  
    //Limpia formulario  
    cedula = "";  
    nombre= "";//2  
    apellido= "";//3  
    sexo= "";//4  
    edad= "";//5  
    document= "";//6  
    identif= "";//7  
    parent= "";//8  
    fecha= null;//9  
    clase= "";//10  
    causa= "";//11  
    observacion= "";//12  
    init();  
  
}
```

```

public List<Ffatipossexos> getIstsexo() {
    return Istsexo;
}

public void setIstsexo(List<Ffatipossexos> Istsexo) {
    this.Istsexo = Istsexo;
}

public List<Ffamortalidaddocumentacion> getIstdocum() {
    return Istdocum;
}

public void setIstdocum(List<Ffamortalidaddocumentacion> Istdocum) {
    this.Istdocum = Istdocum;
}

public List<Ffamortalidadparentesco> getIstparen() {
    return Istparen;
}

public void setIstparen(List<Ffamortalidadparentesco> Istparen) {
    this.Istparen = Istparen;
}

public List<Ffamortalidadclase> getIstclase() {
    return Istclase;
}

public void setIstclase(List<Ffamortalidadclase> Istclase) {
    this.Istclase = Istclase;
}

```

```

        public boolean isHabpaciente() {
            return habpaciente;
        }

        public void setHabpaciente(boolean habpaciente) {
            this.habpaciente = habpaciente;
        }
    }
}

```

Controlador Intervención Familiar

```
package edu.cisc.controlador;
```

```
import java.io.Serializable;
```

```
import java.util.Date;
```

```
import javax.annotation.PostConstruct;
```

```
import javax.faces.application.Application;
```

```
import javax.faces.application.FacesMessage;
```

```
import javax.faces.application.ViewHandler;
```

```
import javax.faces.bean.SessionScoped;
```

```
import javax.faces.component.UIViewRoot;
```

```
import javax.faces.context.FacesContext;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Component;
```

```
import edu.cisc.dao.Ffavivienda_intervencionDao;
```

```
import edu.cisc.modelo.Ffavivienda_intervencion;
```

```
@Component(value="intervencionMB")
```

```
@SessionScoped
```

```
public class DatosIntervencionMB implements Serializable{
```

```

private static final long serialVersionUID = 1L;

//@Autowired(required=true)

private Ffavivienda_intervencion vtabguardar;

//variables para guardar y presentar informacion
private String identif;//1
private String nombre;//2
//private String fecha;//3
private Date fecha;//3
private String cumplimiento;//4
private String resultado;//5
private String acuerdos;//6
private String dificultades;//7
//variable de busqueda de la informacion
private String cedula;
//para validar que se visualice los botones
private boolean habpaciente=false;

@Autowired
private Ffavivienda_intervencionDao vguardar;

@PostConstruct
public void init() {
    //refresh();
}

public void buscarPaciente(){
    FacesContext lContexto= FacesContext.getCurrentInstance();
    Ffavivienda_intervencion buscar = null;

```

```

        if (!(cedula.equals("")) {
            try {
                buscar = vguardar.findbyCedula(cedula);
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
        }

        if (buscar == null) {
            IContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_ERROR, "Error", "Paciente no
Existe"));

            //return "";
        } else if (buscar != null) {
            IContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_INFO, "Información ", "Paciente si
existe en el sistema"));

            //completo=pacienteaut.getPrimernombre()+"
"+pacienteaut.getSegundonombre()+" "+pacienteaut.getApellidoaterno()+"
"+pacienteaut.getApellidomaterno());

            //System.out.println(buscar.getFecha());

            identif=buscar.getIdentificacion();
            nombre=buscar.getNombre_completo();
            fecha=buscar.getFecreg();//este es el q trae de la
base

            cumplimiento=buscar.getCodcumplimiento();
            resultado=buscar.getCodresultado();
            acuerdos=buscar.getDetacuerdos();
            dificultades=buscar.getCausasdificult();
        }

```

```

        cedula="";
        habpaciente=true;
    }

    public void actualizar() {
        try {
            //personaDao.merge(this.nuevaPersona);
            //establecimientosaludDao.merge(estab);
            presentarMensaje("I", "Aplicacion", "Actualizacion exitosa");
        } catch (Exception e) {
            presentarMensaje("E", "Error", "Error en actualizacion:"+e);
        }
    }

    public String guardar(){

        System.out.println("Guardar()");
        int idinterv = 0;
        //Guardar datos en tabla intervencion
        try {

            vtabguardar =
vguardar.guardarVivienda_intervencion(idinterv,//1

            identif,//2

            nombre,//3

            fecha,//4// este es el q guarda a la base

            cumplimiento,//5

            resultado,//6

```

```

        acuerdos,//7

        dificultades//8

    );

    //idinterv = "";//1
    identif= "";//2
    nombre= "";//3
    fecha= null;//4
    cumplimiento= "";//5
    resultado= "";//6
    acuerdos= "";//7
    dificultades= "";//8

    refresh();

    System.out.println("vtabguardar()" );

} catch (Exception e1) {e1.printStackTrace();}

return "";//redireccionar;
}

public String getIdentif() {
    return identif;
}

public void setIdentif(String identif) {
    this.identif = identif;
}

public String getNombre() {

```

```
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public void setFecreg(java.util.Date fecha) {
        this.fecha = (Date) fecha;
    }

    public String getCumplimiento() {
        return cumplimiento;
    }

    public void setCumplimiento(String cumplimiento) {
        this.cumplimiento = cumplimiento;
    }

    public String getResultado() {
        return resultado;
    }

    public void setResultado(String resultado) {
        this.resultado = resultado;
    }

    public String getAcuerdos() {
        return acuerdos;
    }

    public void setAcuerdos(String acuerdos) {
        this.acuerdos = acuerdos;
    }
}
```

```

public String getDificultades() {
    return dificultades;
}

public void setDificultades(String dificultades) {
    this.dificultades = dificultades;
}

public String getCedula() {
    return cedula;
}

public void setCedula(String cedula) {
    this.cedula = cedula;
}

public Date getFecha() {
    return fecha;
}

public void setFecha(Date fecha) {
    this.fecha = fecha;
}

public void presentarMensaje(String severity, String title, String
aplicacion) {
    FacesContext context = FacesContext.getCurrentInstance();
    if(severity.equals("I"))
        context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_INFO,title, aplicacion));
    if(severity.equals("W"))

```

```

        context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_WARN,title, aplicacion));
        if(severity.equals("E"))
            context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_ERROR,title, aplicacion));
        if(severity.equals("F"))
            context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_FATAL,title, aplicacion));

    }

    public void refresh(){

        FacesContext context = FacesContext.getCurrentInstance();
        Application application = context.getApplication();
        ViewHandler viewHandler = application.getViewHandler();
        UIViewRoot viewRoot = viewHandler.createView(context,
context.getViewRoot().getViewId());
        context.setViewRoot(viewRoot);
        context.renderResponse();
    }

    public void limpiar()
    {
        //Método que limpia el formulario en pantalla
        habpaciente=false;

        //Limpia formulario
        cedula = "";
        identif= "";//2
        nombre= "";//3
        fecha= null;//4
        cumplimiento= "";//5
        resultado= "";//6
    }

```

```

        acuerdos= "";//7
        dificultades= "";//8

    }

    public boolean isHabpaciente() {
        return habpaciente;
    }

    public void setHabpaciente(boolean habpaciente) {
        this.habpaciente = habpaciente;
    }
}

```

Controlador Diagnostico Familiar

```

package edu.cisc.controlador;

import java.io.Serializable;
import java.io.IOException;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Date;
import java.util.GregorianCalendar;
import java.util.LinkedHashMap;

import javax.annotation.PostConstruct;
import javax.faces.application.Application;
import javax.faces.application.FacesMessage;
import javax.faces.application.ViewHandler;
import javax.faces.bean.SessionScoped;

```

```

import javax.faces.bean.ViewScoped;
import javax.faces.component.UIViewRoot;
import javax.faces.context.FacesContext;

import javax.persistence.NoResultException;
import javax.faces.context.ExternalContext;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Qualifier;
import org.springframework.stereotype.Component;

import edu.cisc.dao.FfaviviendadiagclaseDao;
import edu.cisc.dao.Ffavivienda_diagnosticoDao;

import edu.cisc.modelo.Ffaviviendadiagclase;
import edu.cisc.modelo.Ffavivienda_diagnostico;

@Component(value="diagnosticoMB")
@SessionScoped
public class DatosDiagnosticoMB implements Serializable{

    private static final long serialVersionUID = 1L;

    //@Autowired(required=true)

    private Ffaviviendadiagclase vdiagclase;
    private Ffavivienda_diagnostico vtabguardar;

    //visita
    private String identif;//1
    private String nombre;//2
    private String clasif;//3

```

```

private Date fecha;//4
private String probl;//5
private String nombreresp;//6
//visita 1
private String identif1;//1
private String nombre1;//2
private String clasif1;//3
private Date fecha1;//4
private String probl1;//5
private String nombreresp1;//6
//visita 2
private String identif2;//1
private String nombre2;//2
private String clasif2;//3
private Date fecha2;//4
private String probl2;//5
private String nombreresp2;//6
//visita 3
private String identif3;//1
private String nombre3;//2
private String clasif3;//3
private Date fecha3;//4
private String probl3;//5
private String nombreresp3;//6
//para seleccionar la cedula
private String cedula;
//para validar que se visualice los botones
private boolean habpaciente=false;

private List<Ffaviviendadiagclase> Istclasificacion;

@Autowired
private FfaviviendadiagclaseDao diagclaseDao;

```

```

    @Autowired
    private Ffavivienda_diagnosticoDao vguardar;

    @PostConstruct
    public void init() {
        //refresh();
        lstclasificacion = new ArrayList<Ffaviviendadiagclase>();

        registrarOpcionesDiagclase();
    }

    public void buscarPaciente(){

        FacesContext lContexto= FacesContext.getCurrentInstance();
        Ffavivienda_diagnostico buscar = null;

        if (!(cedula.equals("")) {
            try {
                buscar = vguardar.findbyCedula(cedula);
            } catch (Exception e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
        }

        if (buscar == null) {
            lContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_ERROR,"Error","Paciente no
Existe"));

            //return "";
        } else if (buscar != null) {

```

```

        IContexto.addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_INFO, "Información ", "Paciente si
existe en el sistema"));

        //completo=pacienteaut.getPrimernombre()+"
"+pacienteaut.getSegundonombre()+" "+pacienteaut.getApellidopaterno()+"
"+pacienteaut.getApellidomaterno());
        System.out.println(buscar.getFecha());

        identif=buscar.getIdentificacion();
        nombre=buscar.getNombre();
        clasif=buscar.getClasificacion();//4
        fecha=buscar.getFecha();//5
        probl=buscar.getProblema();//6
        nombreresp=buscar.getNombreresponsable();//7
        identif1=buscar.getIdentificacion1();//8
        nombre1=buscar.getNombre1();//9
        clasif1=buscar.getClasificacion1();//10
        fecha1=buscar.getFecha1();//11
        probl1=buscar.getProblema1();//12
        nombreresp1=buscar.getNombreresponsable1();//13
        identif2=buscar.getIdentificacion2();//14
        nombre2=buscar.getNombre2();//15
        clasif2=buscar.getClasificacion2();//16
        fecha2=buscar.getFecha2();//17
        probl2=buscar.getProblema2();//18
        nombreresp2=buscar.getNombreresponsable2();//19
        identif3=buscar.getIdentificacion3();//20
        nombre3=buscar.getNombre3();//21
        clasif3=buscar.getClasificacion3();//22
        fecha3=buscar.getFecha3();//23
        probl3=buscar.getProblema3();//24
        nombreresp3=buscar.getNombreresponsable3();//25
    }

```

```

        cedula=" ";
        habpaciente=true;

        /*}catch (NoResultException e) {
            presentarMensaje("E", "Error", "No se encontro datos para
la busquea ingresada");
            init();

        }/*catch (Exception e) {
            // TODO Auto-generated catch block
            System.out.println("error al consultar datos del paciente");
            e.printStackTrace();
        }*/
    }

    public void registrarOpcionesDiagclase(){
        //String idtipo = 1L;
        try {
            lstclasificacion= diagclaseDao.listarporPadre();//(idtipo);
        } catch (Exception e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    public void actualizar() {
        try {
            //personaDao.merge(this.nuevaPersona);
            //establecimientosaludDao.merge(estab);
            presentarMensaje("I", "Aplicacion", "Actualizacion exitosa");
        } catch (Exception e) {
            presentarMensaje("E", "Error", "Error en actualizacion:"+e);
        }
    }
}

```

```

public String guardar(){

    System.out.println("Guardar()");
    int iddiag = 0;
    //Guardar datos en tabla intervencion
    try {

        vtabguardar =
vguardar.guardarVivienda_diagnostico(iddiag,//1

        identif,//2

        nombre,//3

        clasif,//4

        fecha,//5

        probl,//6

        nombreresp,//7

        identif1,//8

        nombre1,//9

        clasif1,//10

        fecha1,//11

        probl1,//12

        nombreresp1,//13

```

identif2,//14

nombre2,//15

clasif2,//16

fecha2,//17

probl2,//18

nombreresp2,//19

identif3,//20

nombre3,//21

clasif3,//22

fecha3,//23

probl3,//24

nombreresp3//25

);

//Limpia formulario

identif= "";//2

nombre= "";//3

clasif= "";//4

fecha= null;//5

probl= "";//6

nombreresp= "";//7

identif1= "";//8

```

        nombre1= "";//9
        clasif1= "";//10
        fecha1= null;//11
        probl1= "";//12
        nombreresp1= "";//13
        identif2= "";//14
        nombre2= "";//15
        clasif2= "";//16
        fecha2= null;//17
        probl2= "";//18
        nombreresp2= "";//19
        identif3= "";//20
        nombre3= "";//21
        clasif3= "";//22
        fecha3= null;//23
        probl3= "";//24
        nombreresp3= "";//25
        //inicializar();
        //inicializarPersona();
        //inicializarUsuario();
        refresh();

        System.out.println("vtabguardar()" );

    } catch (Exception e1) {e1.printStackTrace();}

    return "";//redireccionar;
}

//seguimiento
public String getIdentif() {
    return identif;
}

```

```
public void setIdentif(String identif) {
    this.identif = identif;
}

public String getNombre() {
    return nombre;
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

public String getClasif() {
    return clasif;
}

public void setClasif(String clasif) {
    this.clasif = clasif;
}

public Date getFecha() {
    return fecha;
}

public void setFecha(Date fecha) {
    this.fecha = fecha;
}

public String getProbl() {
    return probl;
}

public void setProbl(String probl) {
    this.probl = probl;
}
```

```
}

public String getNombresp() {
    return nombresp;
}

public void setNombresp(String nombresp) {
    this.nombresp = nombresp;
}

//1 seguimiento
public String getIdentif1() {
    return identif1;
}

public void setIdentif1(String identif1) {
    this.identif1 = identif1;
}

public String getNombre1() {
    return nombre1;
}

public void setNombre1(String nombre1) {
    this.nombre1 = nombre1;
}

public String getClasif1() {
    return clasif1;
}

public void setClasif1(String clasif1) {
    this.clasif1 = clasif1;
}
}
```

```
public Date getFecha1() {
    return fecha1;
}

public void setFecha1(Date fecha1) {
    this.fecha1 = fecha1;
}

public String getProbl1() {
    return probl1;
}

public void setProbl1(String probl1) {
    this.probl1 = probl1;
}

public String getNombresp1() {
    return nombresp1;
}

public void setNombresp1(String nombresp1) {
    this.nombresp1 = nombresp1;
}

//2 seguimiento
public String getIdentif2() {
    return identif2;
}

public void setIdentif2(String identif2) {
    this.identif2 = identif2;
}
```

```
public String getNombre2() {
    return nombre2;
}

public void setNombre2(String nombre2) {
    this.nombre2 = nombre2;
}

public String getClasif2() {
    return clasif2;
}

public void setClasif2(String clasif2) {
    this.clasif2 = clasif2;
}

public Date getFecha2() {
    return fecha2;
}

public void setFecha2(Date fecha2) {
    this.fecha2 = fecha2;
}

public String getProbl2() {
    return probl2;
}

public void setProbl2(String probl2) {
    this.probl2 = probl2;
}

public String getNombresp2() {
    return nombresp2;
}
```

```
}

public void setNombresp2(String nombresp2) {
    this.nombresp2 = nombresp2;
}

//3 seguimiento
public String getIdentif3() {
    return identif3;
}

public void setIdentif3(String identif3) {
    this.identif3 = identif3;
}

public String getNombre3() {
    return nombre3;
}

public void setNombre3(String nombre3) {
    this.nombre3 = nombre3;
}

public String getClasif3() {
    return clasif3;
}

public void setClasif3(String clasif3) {
    this.clasif3 = clasif3;
}

public Date getFecha3() {
    return fecha3;
}
```

```

public void setFecha3(Date fecha3) {
    this.fecha3 = fecha3;
}

public String getProbl3() {
    return probl3;
}

public void setProbl3(String probl3) {
    this.probl3 = probl3;
}

public String getNombresp3() {
    return nombresp3;
}

public void setNombresp3(String nombresp3) {
    this.nombresp3 = nombresp3;
}

public void presentarMensaje(String severity, String title, String
aplicacion) {
    FacesContext context = FacesContext.getCurrentInstance();
    if(severity.equals("I"))
        context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_INFO,title, aplicacion));
    if(severity.equals("W"))
        context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_WARN,title, aplicacion));
    if(severity.equals("E"))
        context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_ERROR,title, aplicacion));
    if(severity.equals("F"))

```

```

        context.addMessage("somekey", new
FacesMessage(FacesMessage.SEVERITY_FATAL,title, aplicacion));

    }

    public List<Ffaviviendadiagclase> getIstclasificacion() {
        return Istclasificacion;
    }

    public void setIstclasificacion(List<Ffaviviendadiagclase> Istclasificacion)
    {
        this.Istclasificacion = Istclasificacion;
    }

    public String getCedula() {
        return cedula;
    }

    public void setCedula(String cedula) {
        this.cedula = cedula;
    }

    public void refresh(){

        FacesContext context = FacesContext.getCurrentInstance();
        Application application = context.getApplication();
        ViewHandler viewHandler = application.getViewHandler();
        UIViewRoot viewRoot = viewHandler.createView(context,
context.getViewRoot().getViewId());
        context.setViewRoot(viewRoot);
        context.renderResponse();
    }

    public void limpiar()

```

```
{  
    //Método que limpia el formulario en pantalla  
    habpaciente=false;  
  
    //Limpia formulario  
    cedula = "";  
    identif= "";//2  
    nombre= "";//3  
    clasif= "";//4  
    fecha= null;//5  
    probl= "";//6  
    nombreresp= "";//7  
    identif1= "";//8  
    nombre1= "";//9  
    clasif1= "";//10  
    fecha1= null;//11  
    probl1= "";//12  
    nombreresp1= "";//13  
    identif2= "";//14  
    nombre2= "";//15  
    clasif2= "";//16  
    fecha2= null;//17  
    probl2= "";//18  
    nombreresp2= "";//19  
    identif3= "";//20  
    nombre3= "";//21  
    clasif3= "";//22  
    fecha3= null;//23  
    probl3= "";//24  
    nombreresp3= "";//25  
    //inicializar();  
    //inicializarPersona();  
    //inicializarUsuario();  
    //refresh();
```

```

    }

    public boolean isHabpaciente() {
        return habpaciente;
    }

    public void setHabpaciente(boolean habpaciente) {
        this.habpaciente = habpaciente;
    }
}

```

Servicios del Sistema Ficha Familiar

Servicio Datos del Paciente

```

package edu.cisc.dao;

import java.util.Date;

import edu.cisc.modelo.Ffavivienda_ubicacion;

public interface Ffavivienda_ubicacionDao {

    public Ffavivienda_ubicacion finbyCedula(String cedu)throws Exception;

    public Ffavivienda_ubicacion guardarVivienda_ubicacion(int
idviviendubic, //1
        String identificacion,//0
        String nombre_primeroy//2
        String nombre_segundoy//3
        String apellido_paterno,//4
        String apellido_materno,//5
        String establecimiento,//6
        String provincia,//7

```

String cantones,//8
String parroquia,//9
String distrito,//10
String zona,//11
String localidad,//12
String sector,//13
String manzana,//14
String division,//15
String edificio,//16
String viviendanumero,//17
String viviendahogar,//18
String calleprincipal,//19
String callesecundaria,//20
String numcasa,//21
String nombrehabitacional,//22
String idlote,//23
String iddepartamento,//24
String idpiso,//25
String convencional,//26
String celular,//27
String referenciahabit,//28
String viviendaocupacion,//29
String vivientatipos,//30
String viviendaacceso,//31
String vivientecho,//32
String vivierendapiso,//33
String vivierendaparedes,//34
String esttechvivienda,//35
String estpisovivienda,//36
String estparedvivienda,//37
String estocupvivienda,//38
String cantcuartos,//39
String cantdormitorios,//40
String viviendaagua,//41

```

        String viviendaagrec,//42
        String viviendaagtrat,//43
        String viviendash,//44
        String viviendashubic,//45
        String viviendaservducha,//46
        String viviendabasura,//47
        String viviendaalumbrado,//48
        String viviendacocina,//49
        String viviendatransporte,//50
        String viviendatiempo,//51
        Date fecusu,//52
        Date fecmod,//53
        String codusu,//54
        String codmod,//55
        String estado,
        String rmedioambvivienda,//56
        String ranimal,//57
            String rsanitario,//58
            String contentr,//59
            String nombenc,//60
            String codenc,//61
            Date fecha,//62
            String numvis
            ) throws Exception;//56
    }

```

```
package edu.cisc.servicio;
```

```
import java.text.SimpleDateFormat;
```

```
import java.util.Date;
```

```
import javax.persistence.EntityManager;
```

```
import javax.persistence.NoResultException;
```

```

import javax.persistence.PersistenceContext;
import javax.persistence.Query;

import org.springframework.stereotype.Component;
import org.springframework.transaction.annotation.Transactional;

import edu.cisc.dao.Ffavivienda_ubicacionDao;
import edu.cisc.modelo.Ffavivienda_ubicacion;

@Component
@Transactional
public class Ffavivienda_ubicacionService implements
Ffavivienda_ubicacionDao {
    //@PersistenceContext(type=PersistenceContextType.EXTENDED)
    //protected EntityManager entityManager;
    @PersistenceContext(unitName="persistenceUnit")
    protected EntityManager entityManager;

    public Ffavivienda_ubicacion finbyCedula(String cedu) throws Exception
    {

        Query query = entityManager.createQuery("select p from
Ffavivienda_ubicacion p where p.identificacion =
:cedu",Ffavivienda_ubicacion.class);
        query.setParameter("cedu", cedu);
        try{
            return (Ffavivienda_ubicacion) query.getSingleResult();
        }catch(NoResultException e){
            throw new NoResultException("DEMO");
        }
    }
}

```

```

@Transactional
public Ffavivienda_ubicacion guardarVivienda_ubicacion(int
idviviendubic,//1
String identificacion,//0
String nombre_primeroy//2
String nombre_segundoy//3
String apellido_paterno,//4
String apellido_materno,//5
String establecimiento,//6
String provincia,//7
String cantones,//8
String parroquia,//9
String distrito,//10
String zona,//11
String localidad,//12
String sector,//13
String manzana,//14
String division,//15
String edificio,//16
String viviendanumero,//17
String viviendahogar,//18
String calleprincipal,//19
String callesecundaria,//20
String numcasa,//21
String nombrehabitacional,//22
String idlote,//23
String iddepartamento,//24
String idpiso,//25
String convencional,//26
String celular,//27
String referenciahabit,//28
String viviendaocupacion,//29
String viviendatipos,//30
String viviendaacceso,//31

```

String viviendatecho,//32
String viviendapiso,//33
String viviendaparedes,//34
String esttechvivienda,//35
String estpisovivienda,//36
String estparedvivienda,//37
String estocupvivienda,//38
String cantcuartos,//39
String cantdormitorios,//40
String viviendaagua,//41
String viviendaagrec,//42
String viviendaagtrat,//43
String viviendash,//44
String viviendashubic,//45
String viviendaservducha,//46
String viviendabasura,//47
String viviendaalumbrado,//48
String viviendacocina,//49
String viviendatransporte,//50
String viviendatiempo,//51
Date fecusu,//52
Date fecmod,//53
String codusu,//54
String codmod,//55
String estado,
String rmedioambvivienda,//56
String ranimal,//57
 String rsanitario,//58
 String contentr,//59
 String nombenc,//60
 String codenc,//61
 Date fecha,//62
 String numvis) throws Exception {//56

```

SimpleDateFormat formatter = new SimpleDateFormat("dd/MM/yyyy");
Long secuenciaubicacion=(long) 0;

//Obtienen el siguiente numero de la secuencia para usar en el insert
secuenciaubicacion = (Long)entityManager.createQuery("select coalesce(
max(idviviendubic),0) + 1 from Ffavivienda_ubicacion u").getSingleResult();
//System.out.println("idagendainter: " + secuenciaAgenda );

Ffavivienda_ubicacion ffavivienda_ubicacion= new Ffavivienda_ubicacion();

ffavivienda_ubicacion.setIdviviendubic(secuenciaubicacion);//1
ffavivienda_ubicacion.setIdidentificacion(identificacion);//0
ffavivienda_ubicacion.setNombre_primero(nombre_primero);//2
ffavivienda_ubicacion.setNombre_segundo(nombre_segundo);//3
ffavivienda_ubicacion.setApellido_paterno(apellido_paterno);//4
ffavivienda_ubicacion.setApellido_materno(apellido_materno);//5
ffavivienda_ubicacion.setEstablecimiento(establecimiento);//6
ffavivienda_ubicacion.setProvincia(provincia);//7
ffavivienda_ubicacion.setCantones(cantones);//8
ffavivienda_ubicacion.setParroquia(parroquia);//9
ffavivienda_ubicacion.setDistrito(distrito);//10
ffavivienda_ubicacion.setZona(zona);//11
ffavivienda_ubicacion.setLocalidad(localidad);//12
ffavivienda_ubicacion.setSector(sector);//13
ffavivienda_ubicacion.setManzana(manzana);//14
ffavivienda_ubicacion.setDivision(division);//15
ffavivienda_ubicacion.setEdificio(edificio);//16
ffavivienda_ubicacion.setViviendanumero(viviendanumero);//17
ffavivienda_ubicacion.setViviendahogar(viviendahogar);//18
ffavivienda_ubicacion.setCalleprincipal(calleprincipal);//19
ffavivienda_ubicacion.setCallesecundaria(callesecundaria);//20
ffavivienda_ubicacion.setNumcasa(numcasa);//21

```

ffavivienda_ubicacion.setNombrehabitacional(nombrehabitacional);//22
ffavivienda_ubicacion.setIdlote(idlote);//23
ffavivienda_ubicacion.setIddepartamento(iddepartamento);//24
ffavivienda_ubicacion.setIdpiso(idpiso);//25
ffavivienda_ubicacion.setConvencional(convencional);//26
ffavivienda_ubicacion.setCelular(celular);//27
ffavivienda_ubicacion.setReferenciahabit(referenciahabit);//28
ffavivienda_ubicacion.setViviendaocupacion(viviendaocupacion);//29
ffavivienda_ubicacion.setViviendatipos(viviendatipos);//30
ffavivienda_ubicacion.setViviendaacceso(viviendaacceso);//31
ffavivienda_ubicacion.setViviendatecho(viviendatecho);//32
ffavivienda_ubicacion.setViviendapiso(viviendapiso);//33
ffavivienda_ubicacion.setViviendaparedes(viviendaparedes);//34
ffavivienda_ubicacion.setEsttechvivienda(esttechvivienda);//35
ffavivienda_ubicacion.setEstpisovivienda(estpisovivienda);//36
ffavivienda_ubicacion.setEstparedvivienda(estparedvivienda);//37
ffavivienda_ubicacion.setEstocupvivienda(estocupvivienda);//38
ffavivienda_ubicacion.setCantcuartos(cantcuartos);//39
ffavivienda_ubicacion.setCantdormitorios(cantdormitorios);//40
ffavivienda_ubicacion.setViviendaagua(viviendaagua);//41
ffavivienda_ubicacion.setViviendaagrec(viviendaagrec);//42
ffavivienda_ubicacion.setViviendaagtrat(viviendaagtrat);//43
ffavivienda_ubicacion.setViviendash(viviendash);//44
ffavivienda_ubicacion.setViviendashubic(viviendashubic);//45
ffavivienda_ubicacion.setViviendaservducha(viviendaservducha);//46
ffavivienda_ubicacion.setViviendabasura(viviendabasura);//47
ffavivienda_ubicacion.setViviendaalumbrado(viviendaalumbrado);//48
ffavivienda_ubicacion.setViviendacocina(viviendacocina);//49
ffavivienda_ubicacion.setViviendatransporte(viviendatransporte);//50
ffavivienda_ubicacion.setViviendatiempo(viviendatiempo);//51
ffavivienda_ubicacion.setFecusu(fecusu);//52
ffavivienda_ubicacion.setFecmod(fecmod);//53
ffavivienda_ubicacion.setCodusu(codusu);//54
ffavivienda_ubicacion.setCodmod(codmod);//55

```

ffavivienda_ubicacion.setEstado(estado);//56
ffavivienda_ubicacion.setRmedioambvivienda(rmedioambvivienda);
ffavivienda_ubicacion.setRanimal(ranimal);
ffavivienda_ubicacion.setRsanitario(rsanitario);
ffavivienda_ubicacion.setContentr(contentr);
ffavivienda_ubicacion.setNombenc(nombenc);
ffavivienda_ubicacion.setCodenc(codenc);
ffavivienda_ubicacion.setFecha(fecha);
ffavivienda_ubicacion.setNumvis(numvis);

```

```

this.entityManager.persist(ffavivienda_ubicacion);

```

```

return ffavivienda_ubicacion;

```

```

}

```

```

    public EntityManager getEntityManager() {
        return entityManager;
    }

```

```

}

```

```

    public void setEntityManager(EntityManager entityManager) {
        this.entityManager = entityManager;
    }

```

```

}

```

```

}

```

Servicio Mortalidad Familiar

```

package edu.cisc.dao;

```

```

import java.util.Date;

```

```

import edu.cisc.modelo.Ffavivienda_mortalidad;

```

```

public interface Ffavivienda_mortalidadDao {

```

```

    public Ffavivienda_mortalidad finbyCedula(String cedu)throws Exception;

```

```
public Ffavivienda_mortalidad guardarVivienda_mortalidad(int
idviviendamort, //1
```

```
String nombre_completo, //2
```

```
String apellido_completo, //3
```

```
String sexo, //4
```

```
String edad, //5
```

```
String documento, //6
```

```
String identificacion, //7
```

```
String pariente, //8
```

```
Date fecreg, //9
```

```
String clase, //10
```

```
String causa, //11
```

```
String observacion //12
```

```
) throws Exception; //
```

```
}
```

```
package edu.cisc.servicio;
```

```
import java.text.SimpleDateFormat;
```

```
import java.util.Date;
```

```

import javax.persistence.EntityManager;
import javax.persistence.NoResultException;
import javax.persistence.PersistenceContext;
import javax.persistence.Query;

import org.springframework.stereotype.Component;
import org.springframework.transaction.annotation.Transactional;

import edu.cisc.dao.Ffavivienda_mortalidadDao;
import edu.cisc.modelo.Ffavivienda_mortalidad;

@Component
@Transactional
public class Ffavivienda_mortalidadService implements
Ffavivienda_mortalidadDao {
    //@PersistenceContext(type=PersistenceContextType.EXTENDED)
    //protected EntityManager entityManager;
    @PersistenceContext(unitName="persistenceUnit")
    protected EntityManager entityManager;

    public Ffavivienda_mortalidad finbyCedula(String cedu) throws Exception
    {

        Query query = entityManager.createQuery("select p from
Ffavivienda_mortalidad p where p.identificacion =
:cedu",Ffavivienda_mortalidad.class);
        query.setParameter("cedu", cedu);
        try{
            return (Ffavivienda_mortalidad) query.getSingleResult();
        }catch(NoResultException e){
            throw new NoResultException("DEMO");
        }
    }
}

```

```
}
```

```
@Transactional
```

```
public Ffavivienda_mortalidad guardarVivienda_mortalidad(int  
idviviendamort, //1
```

```
String nombre_completo, //2
```

```
String apellido_completo, //3
```

```
String sexo, //4
```

```
String edad, //5
```

```
String documento, //6
```

```
String identificacion, //7
```

```
String pariente, //8
```

```
Date fecreg, //9
```

```
String clase, //10
```

```
String causa, //11
```

```
String observacion //12
```

```
) throws Exception {
```

```
SimpleDateFormat formatter = new  
SimpleDateFormat("dd/MM/yyyy");  
Long secuenciamortalidad=(long) 0;
```

```

        //Obtienen el siguiente numero de la secuencia para usar en el
insert
        secuenciamortalidad = (Long)entityManager.createQuery("select
coalesce( max(idviviendamort),0) + 1 from Ffavivienda_mortalidad
u").getSingleResult();
        System.out.println("idviviendamort: " + secuenciamortalidad );

        Ffavivienda_mortalidad ffavivienda_mortalidad= new
Ffavivienda_mortalidad();

        ffavivienda_mortalidad.setIdviviendamort(secuenciamortalidad);//1

        ffavivienda_mortalidad.setNombre_completo(nombre_completo);//2

        ffavivienda_mortalidad.setApellido_completo(apellido_completo);//3
        ffavivienda_mortalidad.setSexo(sexo);//4
        ffavivienda_mortalidad.setEdad(edad);//5
        ffavivienda_mortalidad.setDocumento(documento);//6
        ffavivienda_mortalidad.setIdentificacion(identificacion);//7
        ffavivienda_mortalidad.setPariente(pariente);//8
        ffavivienda_mortalidad.setFecreg(fecreg);//9
        ffavivienda_mortalidad.setClase(clase);//10
        ffavivienda_mortalidad.setCausa(causa);//11
        ffavivienda_mortalidad.setObservacion(observacion);//12

        this.entityManager.persist(ffavivienda_mortalidad);

        return ffavivienda_mortalidad;
    }

    public EntityManager getEntityManager() {
        return entityManager;
    }

```

```

    }

    public void setEntityManager(EntityManager entityManager) {
        this.entityManager = entityManager;
    }
}

```

Servicio Intervención Familiar

```
package edu.cisc.dao;
```

```
import java.util.Date;
```

```
import edu.cisc.modelo.Ffavivienda_intervencion;
```

```
public interface Ffavivienda_intervencionDao {
```

```

    public Ffavivienda_intervencion findbyCedula(String ced)throws
    Exception;
```

```

    public Ffavivienda_intervencion guardarVivienda_intervencion(int
    idviviendinterv, //1
```

```
        String identificacion, //2
```

```
        String nombre_completo, //3
```

```
        Date fecreg, //4
```

```
        String codcumplimiento, //5
```

```
        String codresultado, //6
```

```
        String detacuertos, //7
```

```

        String causasdificult//8

        //String estado

        ) throws Exception;//56
    }

package edu.cisc.servicio;

import java.text.SimpleDateFormat;
import java.util.Date;

import javax.persistence.EntityManager;
import javax.persistence.NoResultException;
import javax.persistence.PersistenceContext;
import javax.persistence.Query;

import org.springframework.stereotype.Component;
import org.springframework.transaction.annotation.Transactional;

import edu.cisc.dao.Ffavivienda_intervencionDao;
import edu.cisc.modelo.Ffavivienda_intervencion;

@Component
@Transactional
public class Ffavivienda_intervencionService implements
Ffavivienda_intervencionDao {
    //@PersistenceContext(type=PersistenceContextType.EXTENDED)
    //protected EntityManager entityManager;
    @PersistenceContext(unitName="persistenceUnit")
    protected EntityManager entityManager;

```

```

        public Ffavivienda_intervencion findbyCedula(String ced) throws
Exception {
            Query query = entityManager.createQuery("select p from
Ffavivienda_intervencion p where p.identificacion =
:ced",Ffavivienda_intervencion.class);
            query.setParameter("ced", ced);
            try{
                return (Ffavivienda_intervencion) query.getSingleResult();
            }catch(NoResultException e){
                throw new NoResultException("DEMO");
            }
        }
    }
}

```

```

@Transactional
public Ffavivienda_intervencion guardarVivienda_intervencion(int
idviviendinterv, //1

```

```

String identificacion,//2

```

```

String nombre_completo,//3

```

```

Date fecreg,//4

```

```

String codcumplimiento,//5

```

```

String codresultado,//6

```

```

String detacuertos,//7

```

```

String causasdificult//8

```

```

) throws Exception {

```

```

        SimpleDateFormat formatter = new
SimpleDateFormat("dd/MM/yyyy");
        Long secuenciaintervencion=(long) 0;

        //Obtienen el siguiente numero de la secuencia para usar en el
insert
        secuenciaintervencion = (Long)entityManager.createQuery("select
coalesce( max(idviviendinterv),0) + 1 from Ffavivienda_intervencion
u").getSingleResult();
        //System.out.println("idagendainter: " + secuenciaintervencion);

        Ffavivienda_intervencion ffavivienda_intervencion= new
Ffavivienda_intervencion();
        //String fecha = fecreg + " ";

        ffavivienda_intervencion.setIdviviendinterv(secuenciaintervencion);//1
        ffavivienda_intervencion.setIdentificacion(identificacion);//2

        ffavivienda_intervencion.setNombre_completo(nombre_completo);//3
        ffavivienda_intervencion.setFecreg(fecreg);//4

        ffavivienda_intervencion.setCodcumplimiento(codcumplimiento);//5

        ffavivienda_intervencion.setCodresultado(codresultado);//6
        ffavivienda_intervencion.setDetacuerdos(detacuerdos);//7
        ffavivienda_intervencion.setCausasdificult(causasdificult);//8

        this.entityManager.persist(ffavivienda_intervencion);

        return ffavivienda_intervencion;
}

```

```

    public EntityManager getEntityManager() {
        return entityManager;
    }

    public void setEntityManager(EntityManager entityManager) {
        this.entityManager = entityManager;
    }
}

```

Servicio Diagnostico Familiar

```
package edu.cisc.dao;
```

```
import java.util.Date;
```

```
import edu.cisc.modelo.Ffavivienda_diagnostico;
```

```
public interface Ffavivienda_diagnosticoDao {
```

```

    public Ffavivienda_diagnostico findbyCedula(String ced)throws
    Exception;
```

```

    public Ffavivienda_diagnostico guardarVivienda_diagnostico(int
    idviviendiag, //1
```

```
        String identificacion,//2
```

```
        String nombre,//3
```

```
        String clasificacion,//4
```

```
        Date fecha,//5
```

```
        String problema,//6
```

```
        String nombresponsable,//7
```

```
        String identificacion1,//8
```

```
        String nombre1,//9
```

```
String clasificacion1,//10
Date fecha1,//11
String problema1,//12
String nombreresponsable1,//13
String identificacion2,//14
String nombre2,//15
String clasificacion2,//16
Date fecha2,//17
String problema2,//18
String nombreresponsable2,//19
String identificacion3,//20
String nombre3,//21
String clasificacion3,//22
Date fecha3,//23
String problema3,//24
String nombreresponsable3//25
) throws Exception;//56
}
```

```
package edu.cisc.servicio;
```

```
import java.text.SimpleDateFormat;
import java.util.Date;
```

```
import javax.persistence.EntityManager;
import javax.persistence.NoResultException;
import javax.persistence.PersistenceContext;
import javax.persistence.Query;
```

```
import org.springframework.stereotype.Component;
import org.springframework.transaction.annotation.Transactional;
```

```
import edu.cisc.dao.Ffavivienda_diagnosticoDao;
import edu.cisc.modelo.Ffavivienda_diagnostico;
```

```

@Component
@Transactional
public class Ffavivienda_diagnosticoService implements
Ffavivienda_diagnosticoDao {
    //@PersistenceContext(type=PersistenceContextType.EXTENDED)
    //protected EntityManager entityManager;
    @PersistenceContext(unitName="persistenceUnit")
    protected EntityManager entityManager;

    public Ffavivienda_diagnostico findbyCedula(String ced) throws
Exception {
        Query query = entityManager.createQuery("select p from
Ffavivienda_diagnostico p where p.identificacion =
:ced",Ffavivienda_diagnostico.class);
        query.setParameter("ced", ced);
        try{
            return (Ffavivienda_diagnostico) query.getSingleResult();
        }catch(NoResultException e){
            throw new NoResultException("DEMO");
        }
    }
}

```

```

@Transactional
public Ffavivienda_diagnostico guardarVivienda_diagnostico(int
idviviendiag, //1
        String identificacion,//2
        String nombre,//3
        String clasificacion,//4
        Date fecha,//5
        String problema,//6

```

```

String nombreresponsable,//7
String identificacion1,//8
String nombre1,//9
String clasificacion1,//10
Date fecha1,//11
String problema1,//12
String nombreresponsable1,//13
String identificacion2,//14
String nombre2,//15
String clasificacion2,//16
Date fecha2,//17
String problema2,//18
String nombreresponsable2,//19
String identificacion3,//20
String nombre3,//21
String clasificacion3,//22
Date fecha3,//23
String problema3,//24
String nombreresponsable3//25
) throws Exception {//56

```

```

SimpleDateFormat formatter = new
SimpleDateFormat("dd/MM/yyyy");
Long secuenciadiagnostico=(long) 0;

//Obtienen el siguiente numero de la secuencia para usar en el
insert
    secuenciadiagnostico = (Long)entityManager.createQuery("select
coalesce( max(idviviendiag),0) + 1 from Ffavivienda_diagnostico
u").getSingleResult());
//System.out.println("idagendainter: " + secuenciaAgenda );

```

```
Ffavivienda_diagnostico ffavivienda_diagnostico= new  
Ffavivienda_diagnostico();
```

```
ffavivienda_diagnostico.setIdiviviendiag(secuenciadiagnostico);//1  
ffavivienda_diagnostico.setIdentificacion(identificacion);//2  
ffavivienda_diagnostico.setNombre(nombre);//3  
ffavivienda_diagnostico.setClasificacion(clasificacion);//4  
ffavivienda_diagnostico.setFecha(fecha);//5  
ffavivienda_diagnostico.setProblema(problema);//6
```

```
ffavivienda_diagnostico.setNombreresponsable(nombreresponsable);//7  
ffavivienda_diagnostico.setIdentificacion1(identificacion1);//8
```

```
ffavivienda_diagnostico.setNombre1(nombre1);//9  
ffavivienda_diagnostico.setClasificacion1(clasificacion1);//10  
ffavivienda_diagnostico.setFecha1(fecha1);//11  
ffavivienda_diagnostico.setProblema1(problema1);//12
```

```
ffavivienda_diagnostico.setNombreresponsable1(nombreresponsable1);//
```

13

```
ffavivienda_diagnostico.setIdentificacion2(identificacion2);//14  
ffavivienda_diagnostico.setNombre2(nombre2);//15  
ffavivienda_diagnostico.setClasificacion2(clasificacion2);//16  
  
ffavivienda_diagnostico.setFecha2(fecha2);//17  
ffavivienda_diagnostico.setProblema2(problema2);//18
```

```
ffavivienda_diagnostico.setNombreresponsable2(nombreresponsable2);//
```

19

```
ffavivienda_diagnostico.setIdentificacion3(identificacion3);//20  
  
ffavivienda_diagnostico.setNombre3(nombre3);//21  
ffavivienda_diagnostico.setClasificacion3(clasificacion3);//22  
ffavivienda_diagnostico.setFecha3(fecha3);//23
```

```

        ffavivienda_diagnostico.setProblema3(problema3);//24

        ffavivienda_diagnostico.setNombreresponsable3(nombreresponsable3);//
25

        this.entityManager.persist(ffavivienda_diagnostico);

        return ffavivienda_diagnostico;
    }

    public EntityManager getEntityManager() {
        return entityManager;
    }

    public void setEntityManager(EntityManager entityManager) {
        this.entityManager = entityManager;
    }
}

```

Modelos del Sistema Ficha Familiar

Modelo Datos del Paciente

```

package edu.cisc.modelo;

import java.io.Serializable;
import java.util.Date;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Table;
import javax.persistence.*;

```

```

/**
 * The persistent class for the sysadm.ffavivienda_ubicacion database table.
 *
 */
@Entity
@Table(name = "ffavivienda_ubicacion", schema="sysadm")
//@NamedQuery(name="Ffavivienda_ubicacion.findAll", query="select h FROM
ffavivienda_ubicacion h")

public class Ffavivienda_ubicacion implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    private long idviviendubic;//1

    private String identificacion;//2
    private String nombre_primerero;//2
    private String nombre_segundo;//3
    private String apellido_paterno;//4
    private String apellido_materno;//5
    private String establecimiento;//6
    private String provincia;//7
    private String cantones;//8
    private String parroquia;//9
    private String distrito;//10
    private String zona;//11
    private String localidad;//12
    private String sector;//13
    private String manzana;//14
    private String division;//15
    private String edificio;//16
    private String viviendanumero;//17
    private String viviendahogar;//18 numhogar

```

private String calleprincipal;//19
private String callesecundaria;//20
private String numcasa;//21
private String nombrehabitacional;//22
private String idlote;//23
private String iddepartamento;//24
private String idpiso;//25
private String convencional;//26
private String celular;//27
private String referenciahabit;//28
private String viviendaocupacion;//29
private String viviendatipos;//30
private String viviendaacceso;//31
private String viviendatecho;//32
private String viviendapiso;//33
private String viviendaparedes;//34
private String esttechvivienda;//35
private String estpisovivienda;//36
private String estparedvivienda;//37
private String estocupvivienda;//38 estocupacvivienda
private String cantcuartos;//39 cantcuartos
private String cantdormitorios;//40 cantdormitorios
private String viviendaagua;//41
private String viviendaagrec;//42
private String viviendaagtrat;//43
private String viviendash;//44
private String viviendashubic;//45
private String viviendaservducha;//46
private String viviendabasura;//47
private String viviendaalumbrado;//48
private String viviendacocina;//49
private String viviendatransporte;//50
private String viviendatiempo;//51 tiempdesplazam
private Date fecusu;//52 fecusu

```

private Date fecmod;//53 fecmod
private String codusu;//54 codusu
private String codmod;//55 codmod
private String estado;//56 estado
private String rmedioambvivienda;//56
private String ranimal;//57
private String rsanitario;//58
private String contentr;//59
private String nombenc;//60
private String codenc;//61
private Date fecha;//62
private String numvis;//63

//constructor
public Ffavivienda_ubicacion() {
}

//constructor identificador
public Ffavivienda_ubicacion(long idviviendubic) {
    this.idviviendubic = idviviendubic;
}

public Ffavivienda_ubicacion(long idviviendubic, //1
    String identificacion,
        String nombre_primeroy//2
        String nombre_segundoy//3
        String apellido_paterno,//4
        String apellido_materno,//5
        String establecimiento,//6
        String provincia,//7
        String cantones,//8
        String parroquia,//9
        String distrito,//10
        String zona,//11

```

String localidad,//12
String sector,//13
String manzana,//14
String division,//15
String edificio,//16
String viviendanumero,//17
String viviendahogar,//18
String calleprincipal,//19
String callesecundaria,//20
String numcasa,//21
String nombrehabitacional,//22
String idlote,//23
String iddepartamento,//24
String idpiso,//25
String convencional,//26
String celular,//27
String referenciahabit,//28
String viviendaocupacion,//29
String vivientatipos,//30
String viviendaacceso,//31
String vivientecho,//32
String vivierendapiso,//33
String vivierendaparedes,//34
String esttechvivienda,//35
String estpisovivienda,//36
String estparedvivienda,//37
String estocupvivienda,//38
String cantcuartos,//39
String cantdormitorios,//40
String viviendaagua,//41
String viviendaagrec,//42
String viviendaagtrat,//43
String vivierendash,//44
String vivierendashubic,//45

```

        String viviendaservducha,//46
        String vivierendabasura,//47
        String viviendaalumbrado,//48
        String vivierendacocina,//49
        String vivierendatransporte,//50
        String vivierendatiempo,//51
        Date fecusu,//52
        Date fecmod,//53
        String codusu,//54
        String codmod,//55
        String estado,
        String rmedioambvivienda,//56
        String ranimal,//57
            String rsanitario,//58
            String contentr,//59
            String nombenc,//60
            String codenc,//61
            Date fecha,//62
            String numvis)//62)//56
    {
        this.idviviendubic = idviviendubic;//1
        this.identificacion = identificacion;
        this.nombre_primer = nombre_primer;//2
        this.nombre_segundo = nombre_segundo;//3
        this.apellido_paterno = apellido_paterno;//4
        this.apellido_materno = apellido_materno;//5
        this.establecimiento = establecimiento;//6
        this.provincia = provincia;//7
        this.cantones = cantones;//8
        this.parroquia = parroquia;//9
        this.districto = districto;//10
        this.zona = zona;//11
        this.localidad = localidad;//12
        this.sector = sector;//13
    }

```

this.manzana = manzana;//14
this.division = division;//15
this.edificio = edificio;//16
this.viviendanumero = viviendanumero;//17
this.viviendahogar = viviendahogar;//18
this.calleprincipal = calleprincipal;//19
this.callesecundaria = callesecundaria;//20
this.numcasa = numcasa;//21
this.nombrehabitacional = nombrehabitacional;//22
this.idlote = idlote;//23
this.iddepartamento = iddepartamento;//24
this.idpiso = idpiso;//25
this.convencional = convencional;//26
this.celular = celular;//27
this.referenciahabit = referenciahabit;//28
this.viviendaocupacion = viviendaocupacion;//29
this.viviendatipos = viviendatipos;//30
this.viviendaacceso = viviendaacceso;//31
this.viendatecho = viendatecho;//32
this.viviendapiso = viviendapiso;//33
this.viendaparedes = viendaparedes;//34
this.esttechvivienda = esttechvivienda;//35
this.estpisovivienda = estpisovivienda;//36
this.estparedvivienda = estparedvivienda;//37
this.estocupvivienda = estocupvivienda;//38
this.cantcuartos = cantcuartos;//39
this.cantdormitorios = cantdormitorios;//40
this.viviendaagua = viviendaagua;//41
this.viviendaagrec = viviendaagrec;//42
this.viviendaagtrat =viviendaagtrat;//43
this.viendash = viendash;//44
this.viendashubic = viendashubic;//45
this.viendaservducha = viendaservducha;//46
this.viendabasura = viendabasura;//47

```

        this.viviendaalumbrado = viviendaalumbrado;//48
        this.viviendacocina = viviendacocina;//49
        this.viviendatransporte = viviendatransporte;//50
        this.viviendatiempo = viviendatiempo;//51
        this.fecusu = fecusu;//52
        this.fecmod = fecmod;//53
        this.codusu = codusu;//54
        this.codmod = codmod;//55
        this.estado = estado;//56
        this.rmedioambvivienda = rmedioambvivienda;
        this.ranimal = ranimal;
        this.rsanitario = rsanitario;
        this.contentr = contentr;
        this.nombenc = nombenc;
        this.codenc = codenc;
        this.fecha = fecha;
        this.numvis = numvis;
    }

    @Column(name = "idviviendubic", unique = true, nullable = false)
    public Long getIdviviendubic() {
        return idviviendubic;
    }

    public void setIdviviendubic(long idviviendubic) {
        this.idviviendubic = idviviendubic;
    }

    @Column(name = "identificacion", length = 100)
    public String getIdentificacion() {
        return identificacion;
    }
}

```

```

public void setIdentificacion(String identificacion) {
    this.identificacion = identificacion;
}

@Column(name = "nombre_primer", length = 100)
public String getNombre_primer() {
    return nombre_primer;
}

public void setNombre_primer(String nombre_primer) {
    this.nombre_primer = nombre_primer;
}

@Column(name = "nombre_segundo", length = 100)
public String getNombre_segundo() {
    return nombre_segundo;
}

public void setNombre_segundo(String nombre_segundo) {
    this.nombre_segundo = nombre_segundo;
}

@Column(name = "apellido_paterno", length = 100)
public String getApellido_paterno() {
    return apellido_paterno;
}

public void setApellido_paterno(String apellido_paterno) {
    this.apellido_paterno = apellido_paterno;
}

@Column(name = "apellido_materno", length = 100)
public String getApellido_materno() {

```

```
        return apellido_materno;
    }

    public void setApellido_materno(String apellido_materno) {
        this.apellido_materno = apellido_materno;
    }

    //
    public String getEstablecimiento() {
        return establecimiento;
    }

    public void setEstablecimiento(String establecimiento) {
        this.establecimiento = establecimiento;
    }

    public String getProvincia() {
        return provincia;
    }

    public void setProvincia(String provincia) {
        this.provincia = provincia;
    }

    public String getCantones() {
        return cantones;
    }

    public void setCantones(String cantones) {
        this.cantones = cantones;
    }

    public String getParroquia() {
        return parroquia;
    }
}
```

```
}

public void setParroquia(String parroquia) {
    this.parroquia = parroquia;
}

public String getDistrito() {
    return distrito;
}

public void setDistrito(String distrito) {
    this.distrito = distrito;
}

public String getZona() {
    return zona;
}

public void setZona(String zona) {
    this.zona = zona;
}

public String getLocalidad() {
    return localidad;
}

public void setLocalidad(String localidad) {
    this.localidad = localidad;
}

public String getSector() {
    return sector;
}
```

```
public void setSector(String sector) {
    this.sector = sector;
}

public String getManzana() {
    return manzana;
}

public void setManzana(String manzana) {
    this.manzana = manzana;
}

public String getDivision() {
    return division;
}

public void setDivision(String division) {
    this.division = division;
}

public String getEdificio() {
    return edificio;
}

public void setEdificio(String edificio) {
    this.edificio = edificio;
}

public String getViviendanumero() {
    return viviendanumero;
}

public void setViviendanumero(String viviendanumero) {
    this.viviendanumero = viviendanumero;
}
```

```
}

public String getViviendahogar() {
    return viviendahogar;
}

public void setViviendahogar(String viviendahogar) {
    this.viviendahogar = viviendahogar;
}

public String getCalleprincipal() {
    return calleprincipal;
}

public void setCalleprincipal(String calleprincipal) {
    this.calleprincipal = calleprincipal;
}

public String getCallesecundaria() {
    return callesecundaria;
}

public void setCallesecundaria(String callesecundaria) {
    this.callesecundaria = callesecundaria;
}

public String getNumcasa() {
    return numcasa;
}

public void setNumcasa(String numcasa) {
    this.numcasa = numcasa;
}
```

```
public String getNombrehabitacional() {
    return nombrehabitacional;
}

public void setNombrehabitacional(String nombrehabitacional) {
    this.nombrehabitacional = nombrehabitacional;
}

public String getIdlote() {
    return idlote;
}

public void setIdlote(String idlote) {
    this.idlote = idlote;
}

public String getIddepartamento() {
    return iddepartamento;
}

public void setIddepartamento(String iddepartamento) {
    this.iddepartamento = iddepartamento;
}

public String getIdpiso() {
    return idpiso;
}

public void setIdpiso(String idpiso) {
    this.idpiso = idpiso;
}

public String getConvencional() {
    return convencional;
}
```

```
}

public void setConvencional(String convencional) {
    this.convencional = convencional;
}

public String getCelular() {
    return celular;
}

public void setCelular(String celular) {
    this.celular = celular;
}

public String getReferenciahabit() {
    return referenciahabit;
}

public void setReferenciahabit(String referenciahabit) {
    this.referenciahabit = referenciahabit;
}

public String getViviendaocupacion() {
    return viviendaocupacion;
}

public void setViviendaocupacion(String viviendaocupacion) {
    this.viviendaocupacion = viviendaocupacion;
}

public String getViviendatipos() {
    return viviendatipos;
}
```

```
public void setViviendatipos(String viviendatipos) {
    this.viviendatipos = viviendatipos;
}

public String getViviendaacceso() {
    return viviendaacceso;
}

public void setViviendaacceso(String viviendaacceso) {
    this.viviendaacceso = viviendaacceso;
}

public String getViviendatecho() {
    return viviendatecho;
}

public void setViviendatecho(String viviendatecho) {
    this.viviendatecho = viviendatecho;
}

public String getViviendapiso() {
    return viviendapiso;
}

public void setViviendapiso(String viviendapiso) {
    this.viviendapiso = viviendapiso;
}

public String getViviendaparedes() {
    return viviendaparedes;
}

public void setViviendaparedes(String viviendaparedes) {
    this.viviendaparedes = viviendaparedes;
}
```

```
}

public String getEsttechvivienda() {
    return esttechvivienda;
}

public void setEsttechvivienda(String esttechvivienda) {
    this.esttechvivienda = esttechvivienda;
}

public String getEstpisovivienda() {
    return estpisovivienda;
}

public void setEstpisovivienda(String estpisovivienda) {
    this.estpisovivienda = estpisovivienda;
}

public String getEstparedvivienda() {
    return estparedvivienda;
}

public void setEstparedvivienda(String estparedvivienda) {
    this.estparedvivienda = estparedvivienda;
}

public String getEstocupvivienda() {
    return estocupvivienda;
}

public void setEstocupvivienda(String estocupvivienda) {
    this.estocupvivienda = estocupvivienda;
}
```

```
public String getCantcuartos() {
    return cantcuartos;
}

public void setCantcuartos(String cantcuartos) {
    this.cantcuartos = cantcuartos;
}

public String getCantdormitorios() {
    return cantdormitorios;
}

public void setCantdormitorios(String cantdormitorios) {
    this.cantdormitorios = cantdormitorios;
}

public String getViviendaagua() {
    return viviendaagua;
}

public void setViviendaagua(String viviendaagua) {
    this.viviendaagua = viviendaagua;
}

public String getViviendaagrec() {
    return viviendaagrec;
}

public void setViviendaagrec(String viviendaagrec) {
    this.viviendaagrec = viviendaagrec;
}

public String getViviendaagtrat() {
    return viviendaagtrat;
}
```

```

}

public void setViviendaagtrat(String viviendaagtrat) {
    this.viviendaagtrat = viviendaagtrat;
}

public String getViviendash() {
    return viviendash;
}

public void setViviendash(String viviendash) {
    this.viviendash = viviendash;
}

public String getViviendashubic() {
    return viviendashubic;
}

public void setViviendashubic(String viviendashubic) {
    this.viviendashubic = viviendashubic;
}

public String getViviendaservducha() {
    return viviendaservducha;
}

public void setViviendaservducha(String viviendaservducha) {
    this.viviendaservducha = viviendaservducha;
}

public String getViviendabasura() {
    return viviendabasura;
}

```

```
public void setViviendabasura(String viviendabasura) {
    this.viviendabasura = viviendabasura;
}

public String getViviendaalumbrado() {
    return viviendaalumbrado;
}

public void setViviendaalumbrado(String viviendaalumbrado) {
    this.viviendaalumbrado = viviendaalumbrado;
}

public String getViviendacocina() {
    return viviendacocina;
}

public void setViviendacocina(String viviendacocina) {
    this.viviendacocina = viviendacocina;
}

public String getViviendatransporte() {
    return viviendatransporte;
}

public void setViviendatransporte(String viviendatransporte) {
    this.viviendatransporte = viviendatransporte;
}

public String getViviendatiempo() {
    return viviendatiempo;
}

public void setViviendatiempo(String viviendatiempo) {
    this.viviendatiempo = viviendatiempo;
}
```

```

}

//@ManyToOne(fetch = FetchType.LAZY)
//@JoinColumn(name = "fecusu")
public Date getFecusu() {
    return fecusu;
}

public void setFecusu(java.util.Date fecusu2) {
    this.fecusu = (java.sql.Date) fecusu2;
}

//@ManyToOne(fetch = FetchType.LAZY)
//@JoinColumn(name = "fecmod")
public Date getFecmod() {
    return fecmod;
}

public void setFecmod(java.util.Date fecmod2) {
    this.fecmod = (Date) fecmod2;
}

//@ManyToOne(fetch = FetchType.LAZY)
//@JoinColumn(name = "codusu")
public String getCodusu() {
    return codusu;
}

public void setCodusu(String codusu) {
    this.codusu = codusu;
}

//@ManyToOne(fetch = FetchType.LAZY)

```

```

//@JoinColumn(name = "codmod")
public String getCodmod() {
    return codmod;
}

public void setCodmod(String codmod) {
    this.codmod = codmod;
}

@Column(name = "estado", length = 1)
public String getEstado() {
    return this.estado;
}

public void setEstado(String estado) {
    this.estado = estado;
}

public String getRmedioambvivienda() {
    return rmedioambvivienda;
}

public void setRmedioambvivienda(String rmedioambvivienda) {
    this.rmedioambvivienda = rmedioambvivienda;
}

public String getRanimal() {
    return ranimal;
}

public void setRanimal(String ranimal) {
    this.ranimal = ranimal;
}

```

```
public String getRsanitario() {
    return rsanitario;
}

public void setRsanitario(String rsanitario) {
    this.rsanitario = rsanitario;
}

public String getContentr() {
    return contentr;
}

public void setContentr(String contentr) {
    this.contentr = contentr;
}

public String getNombenc() {
    return nombenc;
}

public void setNombenc(String nombenc) {
    this.nombenc = nombenc;
}

public String getCodenc() {
    return codenc;
}

public void setCodenc(String codenc) {
    this.codenc = codenc;
}

public Date getFecha() {
    return fecha;
}
```

```

    }

    public void setFecha(Date fecha) {
        this.fecha = fecha;
    }

    public String getNumvis() {
        return numvis;
    }

    public void setNumvis(String numvis) {
        this.numvis = numvis;
    }
}

```

Modelo Mortalidad Familiar

```
package edu.cisc.modelo;
```

```
import java.io.Serializable;
```

```
import java.util.Date;
```

```
import javax.persistence.Column;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.FetchType;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.JoinColumn;
```

```
import javax.persistence.ManyToOne;
```

```
import javax.persistence.Table;
```

```
/**
```

```
 * The persistent class for the sysadm.ffavivienda_mortalidad database table.
```

```
 *
```

```
 */
```

```

@Entity
@Table(name = "ffavivienda_mortalidad", schema="sysadm")
//@NamedQuery(name="Ffavivienda_mortalidad.findAll", query="select h
FROM ffavivienda_mortalidad h")

```

```

public class Ffavivienda_mortalidad implements Serializable {
    private static final long serialVersionUID = 1L;

```

```

    @Id
    private long idviviendamort;//1

```

```

    private String nombre_completo;//2

```

```

    private String apellido_completo;//3

```

```

    private String sexo;//4

```

```

    private String edad;//5

```

```

    private String documento;//6

```

```

    private String identificacion;//7

```

```

    private String pariente;//8

```

```

    private Date fecreg;//9

```

```

    private String clase;//10

```

```

    private String causa;//11

```

```

    private String observacion;//12

```

```

    //constructor

```

```

    public Ffavivienda_mortalidad() {
    }

```

```

    //constructor identificador

```

```

    public Ffavivienda_mortalidad(long idviviendamort) {
        this.idviviendamort = idviviendamort;
    }

```

```

    public Ffavivienda_mortalidad(long idviviendamort, //1
        String nombre_completo, //2

```

```

        String apellido_completo,//3
        String sexo,//4
        String edad,//5
        String documento,//6
        String identificacion,//7
        String pariente,//8
        Date fecreg, //9
        String clase,//10
        String causa,//11
        String observacion //12
    )
}

    this.idviviendamort = idviviendamort;//1
    this.nombre_completo = nombre_completo;//2
    this.apellido_completo = apellido_completo;//3
    this.sexo = sexo;//4
    this.edad = edad;//5
    this.documento = documento;//6
    this.identificacion = identificacion;//7
    this.pariete = pariente;//8
    this.fecreg = fecreg;//9
    this.clase = clase;//7
    this.causa = causa;//8
    this.observacion = observacion;//9
}

```

```

@Column(name = "idviviendamort", unique = true, nullable = false)
public Long getIdviviendamort() {
    return idviviendamort;
}

```

```

public void setIdviviendamort(long idviviendamort) {
    this.idviviendamort = idviviendamort;
}

```

```
}
```

```
@Column(name = "nombre_completo", length = 100)
```

```
public String getNombre_completo() {
```

```
    return nombre_completo;
```

```
}
```

```
public void setNombre_completo(String nombre_completo) {
```

```
    this.nombre_completo = nombre_completo;
```

```
}
```

```
@Column(name = "apellido_completo", length = 100)
```

```
public String getApellido_completo() {
```

```
    return apellido_completo;
```

```
}
```

```
public void setApellido_completo(String apellido_completo) {
```

```
    this.apellido_completo = apellido_completo;
```

```
}
```

```
@Column(name = "edad", length = 100)
```

```
public String getEdad() {
```

```
    return edad;
```

```
}
```

```
public void setEdad(String edad) {
```

```
    this.edad = edad;
```

```
}
```

```
@Column(name = "sexo", length = 100)
```

```
public String getSexo() {  
    return sexo;  
}
```

```
public void setSexo(String sexo) {  
    this.sexo = sexo;  
}
```

```
@Column(name = "documento", length = 100)  
public String getDocumento() {  
    return documento;  
}
```

```
public void setDocumento(String documento) {  
    this.documento = documento;  
}
```

```
@Column(name = "identificacion", length = 100)  
public String getIdentificacion() {  
    return identificacion;  
}
```

```
public void setIdentificacion(String identificacion) {  
    this.identificacion = identificacion;  
}
```

```
@Column(name = "pariente", length = 100)  
public String getPariente() {  
    return pariente;  
}
```

```
public void setPaciente(String paciente) {  
    this.paciente = paciente;  
}
```

```
public Date getFecreg() {  
    return fecreg;  
}
```

```
public void setFecreg(Date fecreg) {  
    this.fecreg = fecreg;  
}
```

```
@Column(name = "clase", length = 100)  
public String getClase() {  
    return clase;  
}
```

```
public void setClase(String clase) {  
    this.clase = clase;  
}
```

```
@Column(name = "causa", length = 100)  
public String getCausa() {  
    return causa;  
}
```

```
public void setCausa(String causa) {  
    this.causa = causa;  
}
```

```
@Column(name = "observacion", length = 100)  
public String getObservacion() {  
    return observacion;  
}
```

```

    }

    public void setObservacion(String observacion) {
        this.observacion = observacion;
    }
}

```

Modelo Intervención Familiar

```
package edu.cisc.modelo;
```

```
import java.io.Serializable;
```

```
import java.util.Date;
```

```
import javax.persistence.Column;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.Table;
```

```
/**
```

```
 * The persistent class for the sysadm.ffavivienda_intervencion database table.
```

```
 *
```

```
 */
```

```
@Entity
```

```
@Table(name = "ffavivienda_intervencion", schema="sysadm")
```

```
//@NamedQuery(name="Ffavivienda_intervencion.findAll", query="select h
```

```
FROM ffavivienda_intervencion h")
```

```
public class Ffavivienda_intervencion implements Serializable {
```

```
    private static final long serialVersionUID = 1L;
```

```
    @Id
```

```
    private long idviviendinterv;//1
```

```

private String identificacion;//2
private String nombre_completo;//3
//private String fecreg;//4
private Date fecreg;//4
private String codcumplimiento;//5
private String codresultado;//6
private String detacuertos;//7
private String causasdificult;//8

//constructor
public Favivienda_intervencion() {
}

//constructor identificador
public Favivienda_intervencion(long idviviendinterv) {
    this.idviviendinterv = idviviendinterv;
}

public Favivienda_intervencion(long idviviendinterv, //1
    String identificacion,//2
    String nombre_completo,//3
    Date fecreg,//4
    String codcumplimiento,//5
    String codresultado,//6
    String detacuertos,//7
    String causasdificult//8
    //String estado //9
)
{
    this.idviviendinterv = idviviendinterv;//1
    this.identificacion = identificacion;//2
    this.nombre_completo = nombre_completo;//3
    this.fecreg = fecreg;//4
    this.codcumplimiento = codcumplimiento;//5
}

```

```

        this.codresultado = codresultado;//6
        this.detacuerdos = detacuerdos;//7
        this.causasdificult = causasdificult;//8
        //this.estado = estado;//56
    }

    @Column(name = "idviviendinterv", unique = true, nullable = false)
    public Long getIdviviendinterv() {
        return idviviendinterv;
    }

    public void setIdviviendinterv(long idviviendinterv) {
        this.idviviendinterv = idviviendinterv;
    }

    @Column(name = "identificacion", length = 100)
    public String getIdentificacion() {
        return identificacion;
    }

    public void setIdentificacion(String identificacion) {
        this.identificacion = identificacion;
    }

    @Column(name = "nombre_completo", length = 100)
    public String getNombre_completo() {
        return nombre_completo;
    }

    public void setNombre_completo(String nombre_completo) {
        this.nombre_completo = nombre_completo;
    }
}

```

```

/*
public void setFecusu(java.util.Date fecusu2) {
    this.fecusu = (java.sql.Date) fecusu2;
}

public Date getFecmod() {
    return fecmod;
}

public void setFecmod(java.util.Date fecmod2) {
    this.fecmod = (Date) fecmod2;
}
*/

@Column(name = "codcumplimiento", length = 100)
public String getCodcumplimiento() {
    return codcumplimiento;
}

public void setCodcumplimiento(String codcumplimiento) {
    this.codcumplimiento = codcumplimiento;
}

@Column(name = "codresultado", length = 100)
public String getCodresultado() {
    return codresultado;
}

public void setCodresultado(String codresultado) {
    this.codresultado = codresultado;
}

```

```

    @Column(name = "detacuerdos", length = 100)
    public String getDetacuerdos() {
        return detacuerdos;
    }

    public void setDetacuerdos(String detacuerdos) {
        this.detacuerdos = detacuerdos;
    }

    @Column(name = "causasdificult", length = 100)
    public String getCausasdificult() {
        return causasdificult;
    }

    public void setCausasdificult(String causasdificult) {
        this.causasdificult = causasdificult;
    }

    public Date getFecreg() {
        return fecreg;
    }

    public void setFecreg(Date fecreg) {
        this.fecreg = fecreg;
    }
}

```

Modelo Diagnostico Familiar

```
package edu.cisc.modelo;
```

```
import java.io.Serializable;
```

```
import java.util.Date;
```

```
import java.util.List;
```

```
import javax.persistence.*;
```

```

/**
 * The persistent class for the sysadm.ffavivienda_diagnostico database table.
 *
 */
@Entity
@Table(name = "ffavivienda_diagnostico", schema="sysadm")
@NamedQuery(name="Ffavivienda_diagnostico.findAll", query="select h
FROM Ffavivienda_diagnostico h")

public class Ffavivienda_diagnostico implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    private long idviviendiag;//1

    private String identificacion;//2
    private String nombre;//3
    private String clasificacion;//4
    private Date fecha;//5
    private String problema;//6
    private String nombreresponsable;//7
    private String identificacion1;//8
    private String nombre1;//9
    private String clasificacion1;//10
    private Date fecha1;//11
    private String problema1;//12
    private String nombreresponsable1;//13
    private String identificacion2;//14
    private String nombre2;//15
    private String clasificacion2;//16
    private Date fecha2;//17
    private String problema2;//18

```

```

private String nombreresponsable2;//19
private String identificacion3;//20
private String nombre3;//21
private String clasificacion3;//22
private Date fecha3;//23
private String problema3;//24
private String nombreresponsable3;//25

@ManyToOne
//@JoinColumn(name="tipo")
private Ffavivienda_diagnostico ffavivienda_diagnostico;

@OneToMany(mappedBy="ffavivienda_diagnostico")
private List<Ffavivienda_diagnostico> ffavivienda_diagnosticos;

//constructor
public Ffavivienda_diagnostico() {
}

//constructor identificador
public Ffavivienda_diagnostico(long idviviendiag) {
    this.idviviendiag = idviviendiag;
}

public Ffavivienda_diagnostico(long idviviendiag, //1
    String identificacion,//2
    String nombre,//3
    String clasificacion,//4
    Date fecha,//5
    String problema,//6
    String nombreresponsable,//7
    String identificacion1,//8
    String nombre1,//9
    String clasificacion1,//10

```

```

        Date fecha1,//11
        String problema1,//12
        String nombreresponsable1,//13
        String identificacion2,//14
        String nombre2,//15
        String clasificacion2,//16
        Date fecha2,//17
        String problema2,//18
        String nombreresponsable2,//19
        String identificacion3,//20
        String nombre3,//21
        String clasificacion3,//22
        Date fecha3,//23
        String problema3,//24
        String nombreresponsable3//25
    )
}

this.idviviendiag = idviviendiag;//1
this.identificacion = identificacion;//2
this.nombre = nombre;//3
this.clasificacion = clasificacion;//4
this.fecha = fecha;//5
this.problema = problema;//6
this.nombreresponsable = nombreresponsable;//7
this.identificacion1 = identificacion1;//2
this.nombre1 = nombre1;//3
this.clasificacion1 = clasificacion1;//4
this.fecha1 = fecha1;//5
this.problema1 = problema1;//6
this.nombreresponsable1 = nombreresponsable1;//7
this.identificacion2 = identificacion2;//2
this.nombre2 = nombre2;//3
this.clasificacion2 = clasificacion2;//4
this.fecha2 = fecha2;//5

```

```

        this.problema2 = problema2;//6
        this.nombrerresponsable2 = nombrerresponsable2;//7
        this.identificacion3 = identificacion3;//2
        this.nombre3 = nombre3;//3
        this.clasificacion3 = clasificacion3;//4
        this.fecha3 = fecha3;//5
        this.problema3 = problema3;//6
        this.nombrerresponsable3 = nombrerresponsable3;//7
    }

    @Column(name = "idviviendiag", unique = true, nullable = false)
    public Long getIdviviendiag() {
        return idviviendiag;
    }

    public void setIdviviendiag(long idviviendiag) {
        this.idviviendiag = idviviendiag;
    }

    @Column(name = "identificacion", length = 100)
    public String getIdentificacion() {
        return identificacion;
    }

    public void setIdentificacion(String identificacion) {
        this.identificacion = identificacion;
    }

    @Column(name = "nombre_completo", length = 100)
    public String getNombre() {
        return nombre;
    }

```

```
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

public Date getFecha() {
    return fecha;
}

public void setFecha(Date fecha) {
    this.fecha = fecha;
}

public String getClasificacion() {
    return clasificacion;
}

public void setClasificacion(String clasificacion) {
    this.clasificacion = clasificacion;
}

public String getProblema() {
    return problema;
}

public void setProblema(String problema) {
    this.problema = problema;
}

public String getNombreresponsable() {
    return nombreresponsable;
}
```

```
public void setNombreresponsable(String nombreresponsable) {  
    this.nombreresponsable = nombreresponsable;  
}
```

```
public String getIdentificacion1() {  
    return identificacion1;  
}
```

```
public void setIdentificacion1(String identificacion1) {  
    this.identificacion1 = identificacion1;  
}
```

```
public String getNombre1() {  
    return nombre1;  
}
```

```
public void setNombre1(String nombre1) {  
    this.nombre1 = nombre1;  
}
```

```
public String getClasificacion1() {  
    return clasificacion1;  
}
```

```
public void setClasificacion1(String clasificacion1) {  
    this.clasificacion1 = clasificacion1;  
}
```

```
public Date getFecha1() {  
    return fecha1;  
}
```

```
public void setFecha1(Date fecha1) {  
    this.fecha1 = fecha1;  
}
```

```
}

public String getProblema1() {
    return problema1;
}

public void setProblema1(String problema1) {
    this.problema1 = problema1;
}

public String getNombrerresponsable1() {
    return nombrerresponsable1;
}

public void setNombrerresponsable1(String nombrerresponsable1) {
    this.nombrerresponsable1 = nombrerresponsable1;
}

public String getIdentificacion2() {
    return identificacion2;
}

public void setIdentificacion2(String identificacion2) {
    this.identificacion2 = identificacion2;
}

public String getNombre2() {
    return nombre2;
}

public void setNombre2(String nombre2) {
    this.nombre2 = nombre2;
}
```

```
public String getClasificacion2() {
    return clasificacion2;
}

public void setClasificacion2(String clasificacion2) {
    this.clasificacion2 = clasificacion2;
}

public Date getFecha2() {
    return fecha2;
}

public void setFecha2(Date fecha2) {
    this.fecha2 = fecha2;
}

public String getProblema2() {
    return problema2;
}

public void setProblema2(String problema2) {
    this.problema2 = problema2;
}

public String getNombreresponsable2() {
    return nombreresponsable2;
}

public void setNombreresponsable2(String nombreresponsable2) {
    this.nombreresponsable2 = nombreresponsable2;
}

public String getIdentificacion3() {
    return identificacion3;
}
```

```
}

public void setIdentificacion3(String identificacion3) {
    this.identificacion3 = identificacion3;
}

public String getNombre3() {
    return nombre3;
}

public void setNombre3(String nombre3) {
    this.nombre3 = nombre3;
}

public String getClasificacion3() {
    return clasificacion3;
}

public void setClasificacion3(String clasificacion3) {
    this.clasificacion3 = clasificacion3;
}

public Date getFecha3() {
    return fecha3;
}

public void setFecha3(Date fecha3) {
    this.fecha3 = fecha3;
}

public String getProblema3() {
    return problema3;
}
```

```

public void setProblema3(String problema3) {
    this.problema3 = problema3;
}

public String getNombreresponsable3() {
    return nombreresponsable3;
}

public void setNombreresponsable3(String nombreresponsable3) {
    this.nombreresponsable3 = nombreresponsable3;
}

public Ffavivienda_diagnostico getFfavivienda_diagnostico() {
    return this.ffavivienda_diagnostico;
}

public void setFfaviviendahogar(Ffavivienda_diagnostico
ffavivienda_diagnostico) {
    this.ffavivienda_diagnostico = ffavivienda_diagnostico;
}

public List<Ffavivienda_diagnostico> getffadiagnosticoclas() {
    return this.ffavivienda_diagnosticos;
}

public void setFfavivienda_diagnosticos(List<Ffavivienda_diagnostico>
ffavivienda_diagnosticos) {
    this.ffavivienda_diagnosticos = ffavivienda_diagnosticos;
}
}

```

Persistencia de Tablas de BD

```

<?xml version="1.0" encoding="UTF-8"?>
<persistence xmlns="http://xmlns.jcp.org/xml/ns/persistence"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

```

```
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence
http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd"
version="2.1">
```

```
<persistence-unit name="persistenceUnit" transaction-
type="RESOURCE_LOCAL">
  <class>edu.cisc.modelo.ActActividad</class>
  <class>edu.cisc.modelo.ActTipoActividad</class>
  <class>edu.cisc.modelo.AdmAplicacion</class>
  <class>edu.cisc.modelo.AdmHistorialConexion</class>
  <class>edu.cisc.modelo.AdmMenu</class>
  <class>edu.cisc.modelo.AdmModulo</class>
  <class>edu.cisc.modelo.AdmModuloRol</class>
  <class>edu.cisc.modelo.AdmPersonaSistema</class>
  <class>edu.cisc.modelo.AdmRol</class>
  <class>edu.cisc.modelo.AdmUsuario</class>
  <class>edu.cisc.modelo.AtmAgenda</class>
  <class>edu.cisc.modelo.AtmAgendaInterconsulta</class>
  <class>edu.cisc.modelo.AtmAltaAmbulatoria</class>
  <class>edu.cisc.modelo.AtmAntFamRegistro</class>
  <class>edu.cisc.modelo.AtmAntPerItem</class>
  <class>edu.cisc.modelo.AtmAntPerRegistro</class>
  <class>edu.cisc.modelo.AtmAntecedenteFamiliar</class>
  <class>edu.cisc.modelo.AtmAntecedentePersonal</class>
  <class>edu.cisc.modelo.AtmCancer</class>
  <class>edu.cisc.modelo.AtmCancerUbica</class>
  <class>edu.cisc.modelo.AtmDescripcionValorClinico</class>
  <class>edu.cisc.modelo.AtmDiagnostico</class>
  <class>edu.cisc.modelo.AtmDiagnosticoEncPaciente</class>
  <class>edu.cisc.modelo.AtmEdifGeoUbicacion</class>
  <class>edu.cisc.modelo.AtmEgresoHospitalario</class>
  <class>edu.cisc.modelo.AtmEncuentroPaciente</class>
  <class>edu.cisc.modelo.AtmFichaAdmision</class>
  <class>edu.cisc.modelo.AtmInformacionClinica</class>
  <class>edu.cisc.modelo.AtmInterconsulta</class>
  <class>edu.cisc.modelo.AtmNotaEvolucion</class>
  <class>edu.cisc.modelo.AtmOrden</class>
  <class>edu.cisc.modelo.AtmOrdenInterconsulta</class>
  <class>edu.cisc.modelo.AtmRegistroAdmision</class>
  <class>edu.cisc.modelo.AtmRegistroAdmisionFicha</class>
  <class>edu.cisc.modelo.AtmResultadoBinario</class>
  <class>edu.cisc.modelo.AtmResultadoCodificado</class>
  <class>edu.cisc.modelo.AtmResultadoNumerico</class>
  <class>edu.cisc.modelo.AtmResultadoTextoLibre</class>
  <class>edu.cisc.modelo.AtmSector</class>
  <class>edu.cisc.modelo.AtmSigVitalMed</class>
  <class>edu.cisc.modelo.AtmTipoConsultaExterna</class>
  <class>edu.cisc.modelo.AtmTipoDiagnostico</class>
  <class>edu.cisc.modelo.AtmTipoInformacionClinica</class>
  <class>edu.cisc.modelo.AtmVisitaHCF</class>
  <class>edu.cisc.modelo.AtmVisitaHCU</class>
  <class>edu.cisc.modelo.EmgIngresoEmergencia</class>
  <class>edu.cisc.modelo.EmgInsumoIngresoEmergencia</class>
  <class>edu.cisc.modelo.EmgTipoEmergencia</class>
  <class>edu.cisc.modelo.EmgTipoTratamiento</class>
  <class>edu.cisc.modelo.EpiCondicionEgreso</class>
  <class>edu.cisc.modelo.EpiCuadroClinico</class>
  <class>edu.cisc.modelo.EpiDiagnostico</class>
  <class>edu.cisc.modelo.EpiEgreso</class>
```

```
<class>edu.cisc.modelo.EpiEvolucionComplicacion</class>
<class>edu.cisc.modelo.EpiHallazgoRelevanteExa</class>
<class>edu.cisc.modelo.EpiMedicoTratante</class>
<class>edu.cisc.modelo.EpiResumenHallazgoClinico</class>
<class>edu.cisc.modelo.FfaActividadEconomica</class>
<class>edu.cisc.modelo.FfaActividadNoTrabajo</class>
<class>edu.cisc.modelo.FfaActividadSemPasada</class>
<class>edu.cisc.modelo.FfaAdulto</class>
<class>edu.cisc.modelo.FfaAgenda</class>
<class>edu.cisc.modelo.FfaCategoriaOcupacion</class>
<class>edu.cisc.modelo.FfaCicloVitalFamiliar</class>
<class>edu.cisc.modelo.FfaCondicionOcupacional</class>
<class>edu.cisc.modelo.FfaDiagnostico</class>
<class>edu.cisc.modelo.FfaEstablecimientoMatricula</class>
<class>edu.cisc.modelo.FfaIntervencion</class>
<class>edu.cisc.modelo.FfaMortalidad</class>
<class>edu.cisc.modelo.FfaMotivoNoMatricula</class>
<class>edu.cisc.modelo.FfaNacionalidad</class>
<class>edu.cisc.modelo.FfaNucleoFamiliar</class>
<class>edu.cisc.modelo.FfaOcupacion</class>
<class>edu.cisc.modelo.FfaParentesco</class>
<class>edu.cisc.modelo.FfaParentescoNucleoFamiliar</class>
<class>edu.cisc.modelo.FfaParticipante</class>
<class>edu.cisc.modelo.FfaPersonaFichaFamiliar</class>
<class>edu.cisc.modelo.FfaTareaAgenda</class>
<class>edu.cisc.modelo.FfaTipoDocumentacion</class>
<class>edu.cisc.modelo.FfaTipoFamilia</class>
<class>edu.cisc.modelo.FfaTipoInstruccion</class>
<class>edu.cisc.modelo.FfaVivienda</class>
<class>edu.cisc.modelo.FfaViviendaControl</class>
<class>edu.cisc.modelo.FfaViviendaDato</class>
<class>edu.cisc.modelo.FfaViviendaFactor</class>
<class>edu.cisc.modelo.Ffacantones</class>
<class>edu.cisc.modelo.Ffadistrito</class>
<class>edu.cisc.modelo.Ffadivision</class>
<class>edu.cisc.modelo.Ffaedificio</class>
<class>edu.cisc.modelo.Ffalocalidad</class>
<class>edu.cisc.modelo.Ffamanzana</class>
<class>edu.cisc.modelo.Ffamortalidadclase</class>
<class>edu.cisc.modelo.Ffamortalidaddocumentacion</class>
<class>edu.cisc.modelo.Ffamortalidadparentesco</class>
<class>edu.cisc.modelo.Ffaparroquia</class>
<class>edu.cisc.modelo.Ffasector</class>
<class>edu.cisc.modelo.Ffatiposexos</class>
<class>edu.cisc.modelo.Ffavivienda_diagnostico</class>
<class>edu.cisc.modelo.Ffavivienda_intervencion</class>
<class>edu.cisc.modelo.Ffavivienda_mortalidad</class>
<class>edu.cisc.modelo.Ffavivienda_ubicacion</class>
<class>edu.cisc.modelo.Ffaviviendaacceso</class>
<class>edu.cisc.modelo.Ffaviviendaagrec</class>
<class>edu.cisc.modelo.Ffaviviendaagtrat</class>
<class>edu.cisc.modelo.Ffaviviendaagua</class>
<class>edu.cisc.modelo.Ffaviviendaalumbrado</class>
<class>edu.cisc.modelo.Ffaviviendabasura</class>
<class>edu.cisc.modelo.Ffaviviendacocina</class>
<class>edu.cisc.modelo.Ffaviviendahogar</class>
<class>edu.cisc.modelo.Ffaviviendadiagclase</class>
<class>edu.cisc.modelo.Ffaviviendaentrevista</class>
<class>edu.cisc.modelo.Ffaviviendaestado</class>
```

```

<class>edu.cisc.modelo.Ffaviviendahogar</class>
<class>edu.cisc.modelo.Ffaviviendanumero</class>
<class>edu.cisc.modelo.Ffaviviendaocupacion</class>
<class>edu.cisc.modelo.Ffaviviendaparedes</class>
<class>edu.cisc.modelo.Ffaviviendapiso</class>
<class>edu.cisc.modelo.Ffaviviendariesgosanimal</class>
<class>edu.cisc.modelo.Ffaviviendariesgosmedio</class>
<class>edu.cisc.modelo.Ffaviviendariesgossanitario</class>
<class>edu.cisc.modelo.Ffaviviendaservducha</class>
<class>edu.cisc.modelo.Ffaviviendash</class>
<class>edu.cisc.modelo.Ffaviviendashubic</class>
<class>edu.cisc.modelo.Ffaviviendatecho</class>
<class>edu.cisc.modelo.Ffaviviendatiempo</class>
<class>edu.cisc.modelo.Ffaviviendatipos</class>
<class>edu.cisc.modelo.Ffaviviendatransporte</class>
<class>edu.cisc.modelo.Ffazona</class>
<class>edu.cisc.modelo.HcfLocalizacion</class>
<class>edu.cisc.modelo.HcfNotificacion</class>
<class>edu.cisc.modelo.HcfOrganizacionSocioEconomica</class>
<class>edu.cisc.modelo.HcfPrincipal</class>
<class>edu.cisc.modelo.HcfRiesgoFamiliar</class>
<class>edu.cisc.modelo.HcfRiesgoSanitarioItem</class>
<class>edu.cisc.modelo.HcfTipoAguaConsumo</class>
<class>edu.cisc.modelo.HcfTipoAlimentacion</class>
<class>edu.cisc.modelo.HcfTipoCalidadVivienda</class>
<class>edu.cisc.modelo.HcfTipoDependenJefeFamilia</class>
<class>edu.cisc.modelo.HcfTipoHijoEducac</class>
<class>edu.cisc.modelo.HcfTipoIngresoFamiliaMensual</class>
<class>edu.cisc.modelo.HcfTipoLuzElectrica</class>
<class>edu.cisc.modelo.HcfTipoMiembroEnfCronica</class>
<class>edu.cisc.modelo.HcfTipoPersonaDormitorio</class>
<class>edu.cisc.modelo.HcfTipoServicioHigienico</class>
<class>edu.cisc.modelo.HcfViviendaServicio</class>
<class>edu.cisc.modelo.ImgCatalogo</class>
<class>edu.cisc.modelo.ImgDatoInforme</class>
<class>edu.cisc.modelo.ImgDatoSolicitud</class>
<class>edu.cisc.modelo.ImgEcografiaObstetrica</class>
<class>edu.cisc.modelo.ImgInforme</class>
<class>edu.cisc.modelo.ImgInformeDiagnostico</class>
<class>edu.cisc.modelo.ImgSolicitud</class>
<class>edu.cisc.modelo.ImgSolicitudDiagnostico</class>
<class>edu.cisc.modelo.ImgTabla</class>
<class>edu.cisc.modelo.LabDescripcionResExamen</class>
<class>edu.cisc.modelo.LabDetalleExamen</class>
<class>edu.cisc.modelo.LabOrdenExamen</class>
<class>edu.cisc.modelo.LabResultadoExamen</class>
<class>edu.cisc.modelo.LabServicioLaboratorio</class>
<class>edu.cisc.modelo.LabServicioLaboratorioDet</class>
<class>edu.cisc.modelo.LprEnfermedad</class>
<class>edu.cisc.modelo.LprExamen</class>
<class>edu.cisc.modelo.LprHistoriaClinicaProblema</class>
<class>edu.cisc.modelo.LprProblema</class>
<class>edu.cisc.modelo.LprProfSaludPaciente</class>
<class>edu.cisc.modelo.LprSegEnfermedad</class>
<class>edu.cisc.modelo.OdoAntecedentePer</class>
<class>edu.cisc.modelo.OdoDiagnostico</class>
<class>edu.cisc.modelo.OdoExaSisEstomatognatico</class>
<class>edu.cisc.modelo.OdoOdontograma</class>
<class>edu.cisc.modelo.OdoPiezaDental</class>

```

```
<class>edu.cisc.modelo.OdoPlanDiagnostico</class>
<class>edu.cisc.modelo.OdoRegistroOdontologico</class>
<class>edu.cisc.modelo.OdoSignoVital</class>
<class>edu.cisc.modelo.OdoSimbologiaOdon</class>
<class>edu.cisc.modelo.OdoTratamiento</class>
<class>edu.cisc.modelo.ParEscolaridad</class>
<class>edu.cisc.modelo.ParEspecialidadMedica</class>
<class>edu.cisc.modelo.ParFacilidad</class>
<class>edu.cisc.modelo.ParFamilia</class>
<class>edu.cisc.modelo.ParGrupoCultural</class>
<class>edu.cisc.modelo.ParHorarioLaboral</class>
<class>edu.cisc.modelo.ParNivelInstruccion</class>
<class>edu.cisc.modelo.ParOrganizacion</class>
<class>edu.cisc.modelo.ParPaciente</class>
<class>edu.cisc.modelo.ParPersona</class>
<class>edu.cisc.modelo.ParPersonaAtmAplicacion</class>
<class>edu.cisc.modelo.ParProfSaludCategoriaServicio</class>
<class>edu.cisc.modelo.ParProfesionalSalud</class>
<class>edu.cisc.modelo.ParPromotor</class>
<class>edu.cisc.modelo.ParTipoEstadoCivil</class>
<class>edu.cisc.modelo.ParTipoIdentificacion</class>
<class>edu.cisc.modelo.ParTipoOrganizacion</class>
<class>edu.cisc.modelo.ParTipoProfesionalSalud</class>
<class>edu.cisc.modelo.ParTipoSeguro</class>
<class>edu.cisc.modelo.ParTipoSexo</class>
<class>edu.cisc.modelo.ParTipoUbicacionGeografica</class>
<class>edu.cisc.modelo.ParUbicacionGeografica</class>
<class>edu.cisc.modelo.ProAdmDetMedicamento</class>
<class>edu.cisc.modelo.ProAdmMedicamento</class>
<class>edu.cisc.modelo.ProCategoriaMedicamento</class>
<class>edu.cisc.modelo.ProCategoriaServicio</class>
<class>edu.cisc.modelo.ProEstablecimientoSalud</class>
<class>edu.cisc.modelo.ProFarmaco</class>
<class>edu.cisc.modelo.ProGrupoTerapeutico</class>
<class>edu.cisc.modelo.ProInsumoMedico</class>
<class>edu.cisc.modelo.ProMedicamento</class>
<class>edu.cisc.modelo.ProMedicamentoSustituto</class>
<class>edu.cisc.modelo.ProProveedor</class>
<class>edu.cisc.modelo.ProServicio</class>
<class>edu.cisc.modelo.ProTipoInsumo</class>
<class>edu.cisc.modelo.ProUnidadOperativa</class>
<class>edu.cisc.modelo.SisMetodoControl</class>
<class>edu.cisc.modelo.SisProcesoHospitalario</class>
<class>edu.cisc.modelo.SviBalanceoHidrico</class>
<class>edu.cisc.modelo.SviMedicionActividad</class>
<class>edu.cisc.modelo.SviSignoVital</class>
<exclude-unlisted-classes>>true</exclude-unlisted-classes>
```

```
</persistence-unit>
```

```
</persistence>
```



UNIVERSIDAD DE GUAYAQUIL

FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS

CARRERA DE INGENIERÍA EN SISTEMAS

COMPUTACIONALES

**“DESARROLLO DEL FORMULARIO ANEXO1-FICHA FAMILIAR SECCIÓN1
DEL MINISTERIO DE SALUD PÚBLICA APLICANDO ARQUETIPOS
BASADO EN NORMA ISO 13606 PARA OBTENER
INTEROPERABILIDAD ENTRE SISTEMAS
HOSPITALARIOS”**

PROYECTO DE TITULACIÓN

Previa la obtención del Título de:

INGENIERO EN SISTEMAS COMPUTACIONALES

AUTORES:

CARLOS ANTONIO MEDINA BRAVO

FERNANDO RICARDO FLORES MOLINA

TUTORA:

ING. MIRIAM ELIZABETH PEÑA GONZÁLEZ MBA.

GUAYAQUIL – ECUADOR

2016



REPOSITORIO NACIONAL EN CIENCIAS Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS

TÍTULO: “Desarrollo del Formulario Anexo1-Ficha Familiar sección 1 del Ministerio de Salud Pública aplicando Arquetipos basado en norma ISO 13606 para obtener interoperabilidad entre sistemas hospitalarios”

REVISORES:

Ing. María Quinzo Bravo MSc.
Ing. Oscar Benavides Burgos MSc.

INSTITUCIÓN:

UNIVERSIDAD DE GUAYAQUIL

FACULTAD:

CIENCIAS MATEMÁTICAS Y FÍSICAS

CARRERA: INGENIERÍA EN SISTEMAS COMPUTACIONALES

FECHA DE PUBLICACIÓN:

Nº DE PÁGS:

124

ÁREA TEMÁTICA: Informática Médica

PALABRAS CLAVES: Arquetipo, CEN/ISO 13606, Historia Clínica Electrónica, Ficha Familiar

RESUMEN: El presente trabajo forma parte del proyecto e-Salud, dedicado a resolver problemas de interoperabilidad que presentan las HCE, mediante uso de arquetipos basado en norma CEN/ISO 13606, se emplean los formularios provistos por el MSP.

Nº DE REGISTRO (en base de datos):

Nº DE CALASIFICACIÓN:

DIRECCIÓN URL (tesis en la web):

ADJUNTO PDF

SI NO

CONTACTO CON AUTORES:

Carlos Antonio Medina Bravo
Fernando Ricardo Flores Molina

Teléfono:

0988465770
0996659803

E-mail:

carlos.medinab@ug.edu.ec
fernando.floresm@ug.edu.ec

CONTACTO CON LA INSTITUCIÓN:

Nombre: Ab. Juan Chávez
Teléfono: 042307729

Aprobación del tutor

En mi calidad de Tutora del trabajo de titulación, “**DESARROLLO DEL FORMULARIO ANEXO 1 - FICHA FAMILIAR SECCIÓN 1 DEL MINISTERIO DE SALUD PÚBLICA APLICANDO ARQUETIPOS BASADO EN NORMA ISO 13606 PARA OBTENER INTEROPERABILIDAD ENTRE SISTEMAS HOSPITALARIOS**”.

Elaborado por el Sr. **CARLOS ANTONIO MEDINA BRAVO** y el Sr. **FERNANDO RICARDO FLORES MOLINA**, alumnos no titulados de la Carrera de Ingeniería en Sistemas Computacionales, Facultad de Ciencias Matemáticas y Físicas de la Universidad de Guayaquil, previo a la obtención del Título de Ingeniero en Sistemas, me permito declarar que luego de haber orientado, estudiado y revisado, la apruebo en sus partes.

Atentamente

ING. MIRIAM ELIZABETH PEÑA GONZÁLEZ MBA

TUTORA

Dedicatoria

Dedico este trabajo principalmente a Dios, a mis abuelos, que han sido mis padres y siempre han estado pendientes de la realización de este trabajo, motivándome día tras día para alcanzar mis metas y objetivos planteados, a mi familia y a verdaderas amistades que han sabido alentarme.

-Carlos Antonio Medina Bravo-

Dedico en primer lugar a Dios por la fuerza que me da para seguir cada día adelante, a mi familia que siempre ha sido mi pilar fundamental en la vida y apoyo para poder culminar mi carrera Universitaria.

-Fernando Ricardo Flores Molina-

Agradecimiento

Doy infinitas gracias a Dios y a mi familia en general porque fueron el motor que me impulsó a terminar un objetivo que comenzó hace varios años en un aula universitaria y a los profesores que supieron guiarme a lo largo de este periodo estudiantil, que termina con la entrega de este proyecto.

-Carlos Antonio Medina Bravo-

Le agradezco en primer lugar a Dios por cada día de vida que me ofrece y las fuerzas de seguir adelante y creciendo profesionalmente. A mis padres que invirtieron en el mejor tesoro que pueden dejar que son los estudios. Mi esposa y mi hija que fueron un apoyo fundamental día a día para lograr esta meta.

-Fernando Ricardo Flores Molina-

Tribunal proyecto de titulación

Ing. Eduardo Santos Baquerizo, MSc.
DECANO DE LA FACULTAD
CIENCIAS MATEMÁTICAS Y
FÍSICAS

Ing. Roberto Crespo Mendoza, Mgs.
DIRECTOR DE LA CARRERA
INGENIERÍA EN SISTEMAS
COMPUTACIONALES

Ing. María Quinzo Bravo, MSc.
PROFESORA REVISORA DEL ÁREA
TRIBUNAL

Ing. Oscar Benavides Burgos, MSc.
PROFESOR REVISOR DEL ÁREA
TRIBUNAL

Ing. Miriam Elizabeth Peña González, Mba.
PROFESORA TUTORA DEL PROYECTO
DE TITULACIÓN

Ab. Juan Chávez Atocha, Esp.
SECRETARIO

Declaración expresa

“La responsabilidad del contenido del presente Proyecto de Titulación, nos corresponde exclusivamente; y el patrimonio intelectual del mismo a la UNIVERSIDAD DE GUAYAQUIL.”

CARLOS ANTONIO MEDINA BRAVO

FERNANDO RICARDO FLORES MOLINA



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

**“Desarrollo del Formulario Anexo1-Ficha Familiar Sección 1 del Ministerio de Salud
Pública aplicando Arquetipos basado en norma ISO 13606 para
obtener interoperabilidad entre sistemas hospitalarios”**

Proyecto de Titulación que se presenta como requisito para optar por el título de
INGENIERO EN SISTEMAS COMPUTACIONALES

Autores: Carlos Antonio Medina Bravo

C.I. 0925595878

Fernando Ricardo Flores Molina

C.I. 0925162265

Tutora: Ing. Miriam Elizabeth Peña González Mba.

Guayaquil, Septiembre del 2016

Certificado de aceptación del tutor

En mi calidad de Tutora del Proyecto de Titulación, nombrada por el Consejo Directivo de la Facultad de Ciencias Matemáticas y Físicas de la Universidad de Guayaquil.

CERTIFICO:

Que he analizado el Proyecto de Titulación presentado por los estudiantes **CARLOS ANTONIO MEDINA BRAVO** y **FERNANDO RICARDO FLORES MOLINA**, como requisito previo para optar por el título de Ingeniero en Sistemas Computacionales cuyo problema es:

“Desarrollo del Formulario Anexo1-Ficha Familiar Sección 1 del Ministerio de Salud Pública aplicando Arquetipos basado en norma ISO 13606 para obtener interoperabilidad entre sistemas hospitalarios”

Presentado por:

MEDINA BRAVO CARLOS ANTONIO

C.I. 0925595878

FLORES MOLINA FERNANDO RICARDO

C.I. 0925162265

Tutora: ING. MIRIAM ELIZABETH PEÑA GONZÁLEZ MBA.

Guayaquil, Septiembre del 2016



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
Autorización para Publicación de Proyecto de Titulación en Formato Digital

1. Identificación del Proyecto de Titulación

Nombre del Alumno: Carlos Antonio Medina Bravo	
Dirección: Calle 22 ava entre calle Q y calle R	
Teléfono: 0988465770	E-mail: carlos.medinab@ug.edu.ec
Nombre del Alumno: Fernando Ricardo Flores Molina	
Dirección: Francisco de Marcos #1915 y José Mascote	
Teléfono: 0996659803	E-mail: fernando.floresm@ug.edu.ec
Facultad: Ciencias Matemáticas y Físicas	
Carrera: Ingeniería en Sistemas Computacionales	
Título al que opta: Ingeniero en Sistemas Computacionales	
Profesora guía: Ing. Miriam Elizabeth Peña González Mba.	
Título del Proyecto de Titulación: “Desarrollo del Formulario Anexo1-Ficha Familiar sección 1 del Ministerio de Salud Pública aplicando Arquetipos basado en norma ISO 13606 para obtener interoperabilidad entre sistemas hospitalarios”	
Temas del Proyecto de Titulación: Sistemas de información, salud, diseño de arquetipos, ISO 13606, interoperabilidad.	

2. Autorización de Publicación de Versión Electrónica del Proyecto de Titulación

A través de este medio autorizo a la Biblioteca de la Universidad de Guayaquil y a la Facultad de Ciencias Matemáticas y Físicas a publicar la versión electrónica de éste Proyecto de Titulación.

Publicación electrónica:

Inmediata	<input checked="" type="checkbox"/>	Después de 1 año	<input type="checkbox"/>
-----------	-------------------------------------	------------------	--------------------------

Firma Alumno:

3. Forma de Envío:

El texto del Proyecto de Titulación debe ser enviado en formato Word, como archivo .Doc. O .RTF y .Puf para PC. Las imágenes que la acompañen pueden ser: .gif, .jpg o .TIFF.

DVDROM	<input checked="" type="checkbox"/>	CDROM	<input type="checkbox"/>
--------	-------------------------------------	-------	--------------------------

Índice general

Aprobación del tutor	III
Dedicatoria.....	IV
Agradecimiento.....	V
Índice general.....	XI
Índice de cuadros	XIV
Índice de gráficos	XV
RESUMEN	XVI
ABSTRACT.....	XVII
1. Introducción	18
2. Capítulo I	20
2.1. El Problema.....	20
2.1.1. Planteamiento del problema.	20
2.1.2. Objetivos.....	25
2.1.3. Alcances del problema.	26
2.1.4. Justificación e Importancia	27
2.1.5. Metodología del proyecto.....	28
2.1.6. Supuestos y Restricciones.	33
3. Capítulo II.....	35
3.1. Marco teórico	35
3.1.1. Antecedentes del estudio	35
3.1.2. Fundamentación teórica.....	36
3.1.3. Estándares ¿por qué?	49
3.1.4. Fundamentación legal.....	69
3.1.5. Pregunta científica a contestarse.....	79
3.1.6. Definiciones conceptuales.	80
4. Capítulo III.....	81
4.1. Propuesta tecnológica.....	81
4.2. Análisis de factibilidad.....	81

4.2.1. Factibilidad operacional.....	82
4.2.2. Factibilidad técnica.....	83
4.2.3. Factibilidad legal.....	85
4.2.4. Factibilidad económica.....	86
4.3. Metodología del proyecto y sus etapas.....	87
4.3.1. Etapas de la Metodología del Proyecto.....	88
4.4. Análisis de requerimientos.....	89
4.4.1. Objetivos de la fase.....	89
4.4.2. Técnicas Utilizadas.....	89
4.5. Entregables del proyecto.....	102
5. Capítulo IV.....	103
5.1. Criterios de aceptación del producto o servicio.....	103
5.2. Conclusiones.....	105
5.3. Recomendaciones.....	106
6. Bibliografía.....	108
7. Anexos.....	112
7.1. Árbol de problemas.....	112
7.2. Árbol de objetivos.....	113
7.3. Formato ficha familiar hoja 1.....	114
7.4. Formato ficha familiar hoja 3.....	115
7.5. Formato ficha familiar hoja 6.....	116
7.6. Formato ficha familiar hoja 8.....	117
7.7. Ejemplo hoja1 sección 1 creando arquetipo.....	118
7.8. Ejemplo Diagrama de Arquetipo General.....	119
7.9. Ejemplo hoja1 sección I árbol de arquetipo.....	120
7.10. Ejemplo hoja1 sección1 vista archivo ADL.....	121
7.11. Ejemplo hoja1 sección1 vista archivo XML.....	122
7.12. Instructivo para manejo de Ficha Familiar.....	123
7.13. Entrevista preguntas de aceptación del proyecto.....	124

Abreviaturas

HC	Historia Clínica
HCU	Historia Clínica Unificada
HCE	Historia Clínica Electrónica
MSP	Ministerio de Salud Pública
HL7	Health Level Seven
UG	Universidad de Guayaquil
ING.	Ingeniero
CC.MM.FF.	Facultad de Ciencias Matemáticas y Físicas
WWW	World Wide Web (red mundial)
ISO	Organización Internacional de Normalización
UNE	Una Norma Española
EN	Norma Europea
URL	Localizador de Fuente Uniforme
INEN	Institución Ecuatoriana de Normalización
MAIS	Modelo de Atención Integral de Salud
DIPA	Departamento de Investigación y Proyectos Académicos
EBAS	Equipos Básicos de Salud
APS	Atención Primaria de Salud
SNS	Sistema Nacional de Salud
EAIS	Equipo de Atención Integral de Salud
TIC's	Tecnologías de Información y Comunicación

Índice de cuadros

Cuadro 1: Causas y Consecuencias del Problema.....	22
Cuadro 2: Delimitación del Problema.....	23
Cuadro 3: Diferencia entre metodologías ágiles y tradicionales.....	32
Cuadro 4: Ventajas e inconvenientes de tipos de aplicaciones.....	46
Cuadro 5: Propiedad intelectual licencia OpenEHR.....	56
Cuadro 6: Características de Hardware.....	84
Cuadro 7: Herramientas de Software implementadas.....	84
Cuadro 8: Costo de proyecto estudiante.....	87
Cuadro 9: Pruebas de funcionalidad.....	103
Cuadro 10: Matriz de criterios de aceptación.....	104

Índice de gráficos

Gráfico 1: Uso de la información de la Ficha Familiar.....	45
Gráfico 2: Modelo típico de tres capas.....	48
Gráfico 3: CEN/ISO 13606 doble modelado.....	51
Gráfico 4: Estructura del modelo de referencia parte 1 y 2.....	52
Gráfico 5: Bloques lógicos de arquetipos.....	53
Gráfico 6: Ejemplo de arquetipo.....	54
Gráfico 7: Modelo de comunicación.....	55
Gráfico 8: Logo OpenEHR.....	55
Gráfico 9: Logo LinkEHR.....	57
Gráfico 10: Principales módulos de la plataforma LinkEHR.....	57



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

“Desarrollo del Formulario Anexo1-Ficha Familiar Sección 1 del Ministerio de Salud Pública aplicando Arquetipos basado en norma ISO 13606 para obtener interoperabilidad entre sistemas hospitalarios”

Autores: Carlos Antonio Medina Bravo
Fernando Ricardo Flores Molina

Tutora: Ing. Miriam Elizabeth Peña González Mba.

RESUMEN

El presente trabajo forma parte de una solución macro presentada como e-Salud, antes llamada Promeinfo, cuenta con la colaboración de las Facultades de Matemática, Medicina y del Departamento de Investigación DIPA de la Universidad de Guayaquil, dedicado a resolver problemas de interoperabilidad que presentan las Historias Clínicas Electrónicas mediante el uso de arquetipos basándose en la norma CEN/ISO 13606 que es un estándar que especifica la estructura y la semántica de un documento clínico con el propósito de lograr un correcto intercambio de la información. Se emplean los formularios básicos provistos por el Ministerio de Salud Pública, de los cuales este proyecto se enfoca en el formulario anexo 1 Ficha Familiar.



UNIVERSIDAD DE GUAYAQUIL
FACULTAD DE CIENCIAS MATEMÁTICAS Y FÍSICAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

"Development of Annex 1 - Sheet Family Form section 1, from Ministry of Public Health using Archetypes based on ISO 13606 for obtain interoperability between hospital systems"

Authors: Carlos Antonio Medina Bravo
Fernando Ricardo Flores Molina

Tutor: Ing. Miriam Elizabeth Peña González Mba.

ABSTRACT

This work is part of a solution macro presented as e-Health, formerly called Promeinfo, with the collaboration of the Faculty of Mathematics, Medicine and the Department of DIPA Research at the University of Guayaquil, dedicated to solving interoperability issues that have clinical Electronic Histories by using archetypes based on CEN / ISO 13606 standard is a standard that specifies the structure and semantics of a clinical document in order to achieve a proper exchange of information. Basic forms provided by the Ministry of Public Health, of which this project focuses on the attached form sheet used 1 Family.

1. Introducción

La Historia Clínica es un documento de vital importancia en todo centro que preste servicios de atención médica. En el que se registra de forma ordenada (clara, precisa, detallada) todos los datos relativos a un enfermo (anteriores y actuales, personales y familiares) que sirven de juicio definitivo de la enfermedad actual o estado de salud actual del paciente. Es el arte de ver, oír, entender y describir la enfermedad humana.

Es un documento legal donde se refleja no solo la práctica médica (todos los procesos al que ha sido sometido un paciente), sino también el cumplimiento de los deberes del personal en salud respecto al paciente, convirtiéndose en la herramienta mediante la cual se mide el nivel de calidad técnico-científica, humana, ética y la responsabilidad del profesional de salud.

Las HC generan diariamente grandes cantidades de información, que deben estar disponibles para ser accedidas cuando se las necesite. Con el avance en Tecnologías de Información de los últimos años, en el campo de medicina muchos establecimientos han optado por el desarrollo y uso de sus propios sistemas médicos que han revolucionado el manejo de información, permitiéndoles ingresar y llevar su historial clínico de una manera más ordenada.

El Ministerio de Salud Pública del Ecuador decide desarrollar y hacer uso de formularios básicos que comprendan una Historia Clínica Única en todos los establecimientos del país, ya que en 1972 a través del decreto supremo N°200 se

obliga el uso de estos formularios. Aunque muchos de los centros médicos tienen sus propios formularios de manera física o electrónica (Ministerio de Salud Pública, 2008).

Después de crearse sistemas médicos que permiten la Historia Clínica Electrónica surge un nuevo problema, que estos sistemas están hechos a medida para cada centro de salud. En países europeos realizaron estudios que puedan solventar este problema mediante el concepto de Arquetipos, creando la norma ISO 13606, cuyo objetivo es dar una estructura informática para comunicar una parte de la HC o su totalidad entre los diferentes centros médicos, obteniendo la tan anhelada Interoperabilidad entre Sistemas Informáticos.

Entre los diferentes Formularios de la HC tenemos el de Ficha Familiar, un instrumento de recolección de información, es el primer contacto entre la familia y el sistema nacional de salud, permite la identificación del grupo familiar, así como la definición de riesgos familiares y seguimiento correspondiente, para saber cuál es la información a captar debemos tener una clara conceptualización de “SALUD”, definiéndola como un estado completo de bienestar biológico, psicológico y social, y no solo como la ausencia de enfermedades (*Ley Orgánica de Salud de la República Del Ecuador, 2006*).

2. Capítulo I

2.1. El Problema

2.1.1. Planteamiento del problema.

2.1.1.1. Ubicación del problema en un contexto.

La falta de interoperabilidad de los sistemas médicos perjudica evidentemente al servicio que brindan las instituciones médicas a sus pacientes, al no tener el control de la información actualizada de cada uno de ellos, no contar con sistemas que permitan intercambiar información de la Historia Clínica (Fuente, 2008).

En muchos de los casos lo que se tiene son islas de información, las mismas que no se encuentran digitalizadas sino que están almacenadas de manera física, lo cual ocasiona la pérdida de información, el desgaste físico (destrucción del documento mediante la manipulación del mismo), además, al momento en que un paciente decida o necesite cambiar de establecimiento médico requerirá nuevamente de un registro de su historial clínico.

La salud del paciente se verá afectada, al no tener su HC actualizada, no se podrá diagnosticar en el tiempo requerido y en el peor de los casos se podría caer en un diagnóstico errado, el riesgo que sufrirá el paciente dependerá de la complejidad o criticidad del estado de salud del mismo. También puede ocasionar que el paciente deba realizarse más de una vez un examen, así como repetir una consulta realizada

con anterioridad en otro centro de salud, estaríamos hablando de pérdida de tiempo, dinero, que podría experimentar el paciente.

Adicionalmente debe recalcar que en el intercambio de la información de la HC del paciente debe mantener su estructura y semántica al pasar entre instituciones médicas, tanto el que emite como el que recibe la información debe ser capaz de poder interpretar lo mismo ya que lo que se está transmitiendo tiene que ver con la salud del paciente y dependerá de un buen diagnóstico médico.

2.1.1.2. Situación Conflicto Nudos Críticos.

El problema se presenta en las diferentes instituciones públicas, privadas, que ofrecen el servicio de salud en el Ecuador, puesto que manejan sus propias reglas de negocio, políticas o estándares referentes al uso de información. Por lo tanto un paciente que asiste por primera vez a estos establecimientos debe llenar una historia clínica, lo cual está correcto, pero en el caso de que no sea su primera visita médica y es en otra institución de salud deberá llenar la Historia Clínica de la institución lo que hace que el paciente debe facilitar nuevamente los datos que se requieren para llenar.

Otro factor que ayuda a generar el problema es que en nuestro país no existe demanda en cuanto a Sistemas de Información que ofrezcan interoperabilidad semántica, lo que ocasiona que se fabriquen solamente sistemas a medida, es decir basado en requerimientos y reglas que establecen los clientes, ocasionando desinterés investigativo sobre normas o estándares que ayuden a los centros de salud a solventar ésta necesidad.

El desconocimiento genera un alto costo de desarrollo ya que implica tiempo de investigación, tanto para la elección de los mejores estándares, como para el desarrollo e implementación de los mismos, llevado de la mano de que no todas estas entidades están dispuestas al cambio y mucho menos a la incertidumbre que generaría una nueva propuesta.

2.1.1.3. Causas y Consecuencias del Problema.

Una de las causas principales es la falta de información de la HC del paciente disponible en el momento en que se requiere, para una oportuna atención y diagnóstico por parte del doctor.

A continuación, se establecen las causas y consecuencias del problema:

Cuadro N° 1 - Causas y consecuencias del problema

CAUSAS	CONSECUENCIAS
En muchos casos no hay información digitalizada, está de manera física.	Desgaste físico o destrucción del documento mediante la manipulación del mismo.
Los establecimientos de Salud utilizan su propio formato de formularios de HC.	A los médicos, no les permite una interpretación automática de la información, sin antes conocer las directrices y/o formato de HC en cada institución médica.
Cada institución médica maneja su propia política referente al uso de información.	No permite intercambiar información clínica con otras instituciones de salud.
Los Sistemas actuales son hechos a medida (según especificaciones de cada institución médica).	En cada institución médica el paciente debe registrar su HC ocasionándole pérdida de tiempo y dinero al tener repetitividad de exámenes y demora de diagnóstico oportuno.

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación, Árbol de problemas

2.1.1.4. Delimitación del Problema.

El cuadro presentado a continuación, contiene la delimitación del problema según el Campo, Aspecto, Tema y Área:

Cuadro N° 2 – Delimitación del problema

ÁREA	Informática Médica (HC)
ASPECTO	Organización, Control y Disponibilidad de la Información Clínica
CAMPO	Salud
TEMA	Desarrollo del Formulario Ficha Familiar del Ministerio de Salud Pública aplicando arquetipo basado en norma ISO 13606 para obtener interoperabilidad entre sistemas hospitalarios

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

2.1.1.5. Formulación del Problema.

¿Cuáles son los factores que imposibilitan a los Centros de Salud del Ecuador tener una mejor administración de la información?

Escasez de digitalización de Historia Clínica en ciertos centros hospitalarios y en otros teniendo HCE, falta de interoperabilidad entre sistemas, con respecto al Formulario Ficha Familiar.

2.1.1.6. Evaluación del Problema.

Los aspectos generales de evaluación son:

Delimitado: Al ser la HC un documento extenso, nuestro trabajo se enfocará en el desarrollo de un formulario en particular llamado ficha familiar, junto con la elaboración de arquetipos que permitan compartir ésta información entre los diferentes establecimientos de salud.

Relevante: La salud de los pacientes es lo más importante, y al lograr interoperabilidad entre los sistemas médicos, se podrá brindar una óptima atención a los pacientes, gracias a la disponibilidad de la información actualizada y confiable.

Evidente: El llenado del formulario ficha familiar se lo hace manualmente, la misma es utilizada por uno o varios médicos y finalmente es archivada, lo que ocasiona el difícil acceso a este documento y posible deterioro por su manipulación en el paso del tiempo.

Concreto: Es una propuesta precisa para lograr una eficiente atención a los pacientes que acuden a las diferentes entidades públicas del país.

Factible: Ya que podemos encontrar información relevante acerca de la norma ISO 13606 junto con la ayuda de herramientas outsourcing, logramos plantear una propuesta que brinde automatización e interoperabilidad de los sistemas médicos.

Original: La elaboración de arquetipos es novedosa en nuestro país, no existen diseños de modelos de información de arquetipos basado en los estándares de la norma ISO 13606.

2.1.2. Objetivos.

2.1.2.1. Objetivo General.

Desarrollar una herramienta tecnológica que permita digitalizar el ingreso de información del formulario Anexo1-Ficha Familiar sección1 utilizada por el MSP para el sistema de información de HC, con el uso de tecnología Open Source a través de arquetipos los cuales están basados en la norma ISO 13606, de manera que se facilite la interoperabilidad de las HC de los usuarios del sistema de salud pública entre las distintas unidades hospitalarias.

2.1.2.2. Objetivos Específicos.

- Diseñar y desarrollar la interface visual que permita la digitalización del formulario de Ficha Familiar establecido por el MSP como parte de la Historia Clínica Electrónica (HCE) el cual será construido con lenguajes de programación y herramientas última generación de tipo Open Source.
- Diseñar y establecer una línea base para la creación de los arquetipos con profesionales médicos en función del formulario de Ficha Familiar, haciendo uso de la Norma ISO 13606, lo cual permitirá facilitar la interoperabilidad de la información entre los diferentes sistemas médicos existentes.
- Implantar la solución tecnológica desarrollada integrándola al Sistema de Histórica Clínica Única.
- Realizar las pruebas de software a nivel local detalladas en un plan de pruebas para verificar y garantizar que el sistema de información funcione correctamente.

2.1.3. Alcances del problema.

Para el levantamiento de información se revisará el manual de uso de los formularios básicos de historia clínica única (HCU) y se realizarán reuniones con doctores de los establecimientos de salud que tienen experiencia en llenado de HC.

Se realizará el análisis de la base de datos entregada por parte de Promeinfo 2, para adaptarla según sea necesario y realizar los ingresos-consultas de la información del formulario, el diseño se lo realizará con el framework de aplicaciones Java/J2EE Spring MVC. Los arquetipos para el formulario Ficha Familiar se desarrollaran bajo la norma ISO 13606 utilizando el programa LinKEHR.

El formulario permitirá el ingreso de la información de las familias, y cada uno de sus miembros.

Se realizarán las pruebas de funcionalidad del formulario por separado e integrado en el sistema de HCU.

2.1.4. Justificación e Importancia.

El presente proyecto forma parte del Sistema HCU, que tiene como objetivo principal la automatización con respecto al manejo de información de un paciente (actualizada, ordenada y confiable), con lo cual podrá recibir una atención médica de forma rápida sin tener que realizar un nuevo levantamiento de Historia Clínica con la nueva ventaja de interoperabilidad mediante el diseño de arquetipos basados en la norma ISO 13606.

Con esto cualquier paciente podrá asistir a los diferentes establecimientos médicos y con su número de identificación podrán visualizar su HCU en donde el médico por medio de la página WEB tendrá disponible la información médica y patológica del paciente y de una mejor manera determinar un diagnóstico y medicamentos al mismo.

El implementar arquetipos basados en la norma ISO 13606 nos permite un crecimiento tecnológico para el país, porque al tratarse de un tema novedoso, provocará mucho interés e iniciativa en cuanto a investigación mediante la elaboración de artículos científicos que beneficiará a la CISC sobre temas de acreditación.

2.1.5. Metodología del proyecto.

2.1.5.1. Metodología de Desarrollo.

La metodología que se eligió para desarrollar este proyecto se deriva de las metodologías ágiles.

“En una reunión celebrada en febrero de 2001 en Utah-EEUU, nace el término ágil aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software” (Letelier & Penadés, 2006).

A continuación enumeraremos las características de la metodología ágil:

- “Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Si se sigue un buen proceso de desarrollo, pero el equipo falla, el éxito no está asegurado; sin embargo, si el equipo funciona, es más fácil conseguir el objetivo final, aunque no se tenga un proceso bien definido. No se necesitan desarrolladores brillantes, sino desarrolladores que se adapten bien al trabajo en equipo. Así mismo, las herramientas (compiladores, depuradores, control de versiones, etc.) son importantes para mejorar el rendimiento del equipo, pero el disponer más recursos que los estrictamente necesarios también puede afectar negativamente. En resumen, es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

- Desarrollar software que funciona más que conseguir una buena documentación. Aunque se parte de la base de que el software sin documentación es un desastre, la regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental. Si una vez iniciado el proyecto, un nuevo miembro se incorpora al equipo de desarrollo, se considera que los dos elementos que más le van a servir para ponerse al día son: el propio código y la interacción con el equipo.
- La colaboración con el cliente más que la negociación de un contrato. Las características particulares del desarrollo de software hace que muchos proyectos hayan fracasado por intentar cumplir unos plazos y unos costes preestablecidos al inicio del mismo, según los requisitos que el cliente manifestaba en ese momento. Por ello, se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta puesto que hay muchas variables en juego, debe ser flexible para poder adaptarse a los cambios que puedan surgir. Una buena estrategia es hacer planificaciones detalladas para unas pocas semanas y planificaciones mucho más abiertas para unos pocos meses.

- Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas. Las entregas al cliente se insisten en que sean software, no planificaciones, ni documentación de análisis o de diseño.
- La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto. El proceso de desarrollo necesita ser guiado por el cliente, por lo que la interacción con el equipo es muy frecuente.
- Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo. La gente es el principal factor de éxito, todo los demás (proceso, entorno, gestión, etc.) queda en segundo plano. Si cualquiera de ellos tiene un efecto negativo sobre los individuos debe ser cambiado.
- El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo. Los miembros de equipo deben hablar entre ellos, éste es el principal modo de comunicación. Se pueden crear documentos pero no todo estará en ellos, no es lo que el equipo espera.
- El software que funciona es la medida principal de progreso. El estado de un proyecto no viene dado por la documentación generada o la fase en la que se encuentre, sino por el código generado y en funcionamiento. Por ejemplo, un proyecto se encuentra al 50% si el 50% de los requisitos ya están en funcionamiento.

- Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante. No se trata de desarrollar lo más rápido posible, sino de mantener el ritmo de desarrollo durante toda la duración del proyecto, asegurando en todo momento que la calidad de lo producido es máxima.
- La atención continua a la calidad técnica y al buen diseño mejora la agilidad. Producir código claro y robusto es la clave para avanzar más rápidamente en el proyecto.
- La simplicidad es esencial. Tomar los caminos más simples que sean consistentes con los objetivos perseguidos. Si el código producido es simple y de alta calidad será más sencillo adaptarlo a los cambios que puedan surgir.
- Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos. Todo el equipo es informado de las responsabilidades y éstas recaen sobre todos sus miembros. Es el propio equipo el que decide la mejor forma de organizarse, de acuerdo a los objetivos que se persigan.
- En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento. Puesto que el entorno está cambiando continuamente, el equipo también debe ajustarse al nuevo escenario de forma continua. Puede cambiar su organización, sus reglas, sus convenciones, sus relaciones, etc., para seguir siendo ágil” (Letelier & Penadés, 2006, págs. 3-4)

- Comparación Metodologías Ágiles y Tradicionales.

El cuadro a continuación se detalla las principales diferencias de una Metodología Ágil respecto de las Metodologías Tradicionales.

Cuadro N° 3 – Diferencia entre metodologías ágiles y tradicionales

Metodología Ágil	Metodología Tradicional
Pocos Artefactos. El modelado es prescindible, modelos desechables.	Más Artefactos. El modelado es esencial, mantenimiento de modelos
Pocos Roles, más genéricos y flexibles	Más Roles, más específicos
No existe un contrato tradicional, debe ser bastante flexible	Existe un contrato prefijado
Cliente es parte del equipo de desarrollo (además in-situ)	El cliente interactúa con el equipo de desarrollo mediante reuniones
Orientada a proyectos pequeños. Corta duración (o entregas frecuentes), equipos pequeños (< 10 integrantes) y trabajando en el mismo sitio	Aplicables a proyectos de cualquier tamaño, pero suelen ser especialmente efectivas/usadas en proyectos grandes y con equipos posiblemente dispersos
La arquitectura se va definiendo y mejorando a lo largo del proyecto	Se promueve que la arquitectura se defina tempranamente en el proyecto
Énfasis en los aspectos humanos: el individuo y el trabajo en equipo	Énfasis en la definición del proceso: roles, actividades y artefactos
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Se esperan cambios durante el proyecto	Se espera que no ocurran cambios de gran impacto durante el proyecto

(Letelier & Penadés, 2006)

Fuente: <http://www.cyta.com.ar/ta0502/v5n2a1.htm>

2.1.6. Supuestos y Restricciones.

2.1.6.1. Supuestos.

- Se dispondrá de un servidor para poder subir la aplicación web y el servicio web.
- La base de datos contendrá toda la información referente a los servicios que brinda el establecimiento de salud.
- Se usará el servidor de aplicaciones tomcat-apache para levantar la aplicación web.

2.1.6.2. Restricciones.

- Capacitación sobre las herramientas que se usarán para el desarrollo.
- Análisis de los resultados del proceso anterior.
- Uso de recursos propios para el desarrollo del proyecto.

2.1.6.3. Plan de Calidad.

Se realiza un plan de pruebas para verificar que el proyecto este operativo, que cumpla con los objetivos, esto es, que se ingresen los datos y que se produzca los resultados esperados.

Se realizarán las siguientes pruebas:

Prueba: Verificar que la funcionalidad del proyecto esté de acuerdo a los requerimientos propuestos.

Método: Se procede a ingresar información sea esta valida o no.

Resultados: Si la información es válida, se procederá a guardar y posteriormente se podrá consultar y si la información no es válida entonces deberá mostrar un mensaje con el error que se produjo.

Prueba: Verificar que el diseño del proyecto funcione en diferentes ambientes.

Método: Ejecutar la aplicación web en los diferentes sistemas operativos.

Resultados: La aplicación se ejecute correctamente.

Prueba: Verificar que el diseño del proyecto funcione en los diferentes navegadores web.

Método: Ejecutar la aplicación web en los diferentes navegadores existentes.

Resultados: El diseño no se vea afectado y sus componentes se muestren correctamente en el navegador.

3. Capítulo II

3.1. Marco teórico

3.1.1. Antecedentes del estudio

La historia clínica es considerada un instrumento médico legal donde vemos reflejado una relación médico-paciente.

El Ministerio de Salud Pública fue creado el 16 de Junio de 1967 por mandato de la Asamblea Constituyente, asumiendo el control de las entidades públicas de salud, en ese entonces ya existían formularios de historia clínica pero eran diferentes ya que fueron diseñados de acuerdo a los criterios de los distintos servicios de salud definidos por estas entidades de salud (Ministerio de Salud Pública, 2008).

Mediante el Decreto Supremo N° 200 en 1972 el organismo asesor del MSP compuesto por el Comité Nacional de Coordinación y Desarrollo de las Estadísticas de Salud y de Historia Clínica, impuso el uso obligatorio a las entidades de salud pública de Historia Clínica Única, como un grupo de formularios y métodos de registro médico de acuerdo con los procedimientos dictados por el MSP (Ministerio de Salud Pública, 2008).

Durante el paso de los años existieron muchas propuestas diferentes de los modelos de los formularios, rediseñándolos para su presentación ante el Consejo Nacional de

Salud (CONASA), Pero en octubre del año 2006 el Directorio de CONASA aprobó 14 formularios básicos para el expediente único de historia clínica.

En enero del 2007, mediante Acuerdo Ministerial se aprobó la utilización de los formularios básicos actualizados de la Historia Clínica Única (...) En febrero de 2007, la Comisión Ministerial de la Historia Clínica, con apoyo técnico de OPS, consolidó los aportes recopilados en la fase de implementación – evaluación y presentó 20 formularios básicos para su aprobación y aplicación. (Chang Campos, Revista Judicial , 2008)

Como tradición se ha utilizado la Historia Clínica en papel y al pasar de los años ha tenido un crecimiento de los archivos de la misma obligando a situar dichos archivos en lugares externos al centro de salud. Esto permitió la aparición de la HCE que reduce enormemente las necesidades de recurso humano y espacio físico para disponer de las HC en papel.

3.1.2. Fundamentación teórica

En esta parte encontraremos definiciones necesarias para poder explicar de mejor manera el desarrollo de este proyecto.

E-Salud anteriormente conocido como Promeinfo es un programa continuo de Investigación médico-científica de la Universidad de Guayaquil. Es multidisciplinario, multifacultades, que contemplan una diversidad de trabajos de investigación, desarrollo e innovación en los campos de informática médica.

Este programa forma parte de los Proyectos Emblemáticos con el Fondo Competitivo de Investigación (FCI) de la Dirección de Investigación y Proyectos Académicos (DIPA) de la Universidad de Guayaquil.

3.1.2.1. Ministerio de salud pública del Ecuador.

El MSP tiene como misión ejercer la rectoría, regulación, planificación, coordinación, control y gestión de la salud pública a través de la gobernanza, vigilancia, control sanitario y garantizar el derecho a la salud a través de la provisión de servicios de atención individual, prevención de enfermedades, promoción de la salud e igualdad, la gobernanza de salud, investigación y desarrollo de la ciencia y tecnología; articulación de los actores del sistema, con el fin de garantizar el derecho a la Salud (Ministerio de Salud Pública, s. f.).

La Visión del MSP es, ejercer plenamente la gobernanza del Sistema Nacional de Salud, con un modelo referencial en Latinoamérica que priorice la promoción de la salud y la prevención de enfermedades, con altos niveles de atención de calidad, con calidez, garantizando la salud integral de la población y el acceso universal a una red de servicios, con la participación coordinada de organizaciones públicas, privadas y de la comunidad (Ministerio de Salud Pública, s. f.)

El MSP (2016) en su página web muestra como valores los siguientes:

- **Respeto.-** Entendemos que todas las personas son iguales y merecen el mejor servicio, por lo que nos comprometemos a respetar su dignidad y a atender sus necesidades teniendo en cuenta, en todo momento, sus derechos.
- **Inclusión.-** Reconocemos que los grupos sociales son distintos y valoramos sus diferencias.
- **Vocación de servicio.-** Nuestra labor diaria lo hacemos con pasión.
- **Compromiso.-** Nos comprometemos a que nuestras capacidades cumplan con todo aquello que se nos ha confiado.
- **Integridad.-** Tenemos la capacidad para decidir responsablemente sobre nuestro comportamiento”.
- **Justicia.-** Creemos que todas las personas tienen las mismas oportunidades y trabajamos para ello.
- **Lealtad.-** Confianza y defensa de los valores, principios y objetivos de la entidad, garantizando los derechos individuales y colectivos.(p. 1)

3.1.2.2. Modelo de atención integral de salud.

El Modelo de Atención Integral de Salud (MAIS) es un conjunto de políticas, estrategias, lineamientos y herramientas enfocadas en la salud Familiar, Comunitaria e Intercultural que complementándose, organiza y descongestiona el Sistema Nacional de Salud, permitiendo la integralidad de los tres niveles de atención en la red de salud. (MSP, 2012)

El MAIS define la interacción de los sectores públicos y privados, los miembros de la red de servicios de salud y la comunidad para llevar a cabo acciones conjuntas que permitan dar soluciones integrales a las necesidades y problemas de la comunidad.

Además, reorienta los servicios de salud hacia la promoción de la salud y prevención de la enfermedad con la participación organizada de los sujetos sociales, también fortalece la curación, recuperación, rehabilitación de la salud para brindar una atención integral, de calidad y de profundo respeto a las personas en su diversidad y entorno. (MSP, 2012)

El MSP (2012) en su artículo define los objetivos que persigue el MAIS son cinco:

Reorientar los servicios de salud del enfoque curativo hacia un enfoque centrado en la promoción y cuidado integral de la salud, garantizado una respuesta en el proceso de recuperación de la salud, rehabilitación, cuidados paliativos; incorporando los enfoques de interculturalidad, generacional y de género. Fortalecimiento de la vigilancia epidemiológica comunitaria.

Implementar estrategias para el desarrollo y la gestión del talento humano a través de procesos de capacitación continua, la carrera sanitaria y la formación en los servicios. En el primero y segundo nivel se incorpora especialistas en Medicina Familiar y Comunitaria y Técnicos de APS.

Organizar los servicios del sector público conforme los estándares establecidos por el Ministerio de Salud Pública.

Organizar el Sistema Único de Información en Salud en Coordinación con Sistema Común de Información del Ministerio de Salud Pública.

Fortalecer la participación plena de la comunidad y de los colectivos organizados. (pág. 1)

3.1.2.3. Equipos básicos de salud.

Los Equipos Básicos de Salud (EBAS) es una de las estrategias del (MAIS) a nivel Familiar, Comunitario e Intercultural a cargo del (MSP).

Cada equipo bajo la denominación de EBAS para prestación de sus servicios en las áreas de salud, conformado por un médico, una obstetra, una enfermera, un odontólogo y una auxiliar de enfermería trabaja exclusivamente en los objetivos y metas del modelo. Este equipo tiene señaladas actividades integrales preventivas y curativas con un amplio margen, para de acuerdo a cada diagnóstico establecer las estrategias más apropiadas, remitiendo los problemas de mayor complejidad, participación con la comunidad y sus representantes en procesos de planificación y evaluación, diseño y ejecución de propuestas educativas. (SIISE, s. f., pág. 1)

Cada EBAS es responsable de:

- 500 a 600 familias en el área rural
- 900 a 1000 familias en el área urbana y urbana marginal.

El tipo de información que receptan sirve para que los sub-centros y áreas de salud de cada provincia del país, obtengan información de primera mano respecto a la salud de cada uno de los habitantes. Esta estrategia durante su ejecución 2008-2010, permitió la interrelación con comunidades dispersas brindando educación útil en salud y así educar gente con calidez y calidad. (SIISE, s. f.)

3.1.2.4. Sistemas de información.

El término sistemas se lo emplea mayormente para referirnos a diferentes y variadas situaciones como: sistemas eléctricos, sistemas monetarios, sistemas bancarios, etc. Comúnmente se lo emplea para categorizar una definición o una herramienta para explicar cómo funciona y que acontezca en una determinada área mencionada anteriormente como son la economía, tecnología, física, etc.

Podemos definir que un sistema es un conjunto de elementos relacionados que interactúan de una forma ordenada para lograr un objetivo. Existen muchos conceptos de sistemas de información, y algunos autores lo enfocan hacia los sistemas empresariales definiéndolos como “un conjunto de recursos técnicos humanos y económicos, interrelacionados dinámicamente, y organizados en torno al objetivo de satisfacer la necesidades de información de una organización empresarial para la gestión y la correcta adopción de decisiones” (De Pablos Heredero, López Hermoso Agius, Romo Romero, & Medina Salgado, 2011, pág. 20).

3.1.2.5. Historia clínica.

En su estudio (Guzmán & Arias, 2012) Afirmaron que:

La historia clínica es una de las formas de registro del acto médico, cuyas cuatro características principales se encuentran involucradas en su elaboración y son: profesionalidad, ejecución típica, objetivo y licitud. La profesionalidad se refiere a que solamente el profesional de la medicina puede efectuar un acto médico, pues en esencia son los médicos quienes están en capacidad de elaborar una buena historia clínica. La ejecución es típica cuando se hace conforme a la denominada *lex artis ad hoc*, debido a que la medicina siempre se ejerce de acuerdo con las normas de excelencia de ese momento, a pesar de las limitaciones de tiempo, lugar y entorno. El objetivo de ayuda al enfermo se traduce en aquello que se transcribe en la historia. La licitud se debe a que la misma norma jurídica respalda a la historia clínica como documento indispensable (p.15).

Podemos definir a la HC como un documento que se elabora al momento que existe un acercamiento entre un médico y un paciente, donde se registra la información indispensable para poder darle una mejor atención al paciente.

La HC, según la define (HERRERA RAMIREZ, 2008) en su libro Manual de responsabilidad médica, es: El documento o conjunto de folios, anotaciones, registros, información exámenes, descripciones médicas, formulaciones, en fin, es la recopilación escrita de toda información, de todo lo ocurrido que hay que llevar desde el inicio del ingreso del paciente a la institución, hasta su salida (p. 69).

“Es un documento confidencial y obligatorio de carácter técnico y legal, compuesto por un conjunto de formularios básicos y de especialidad, que el personal de la salud utiliza para registrar en forma sistemática los datos obtenidos

de las atenciones, diagnóstico, tratamiento, evolución y resultados de salud y enfermedad durante todo el ciclo vital del/la usuario/a” (Registro Oficial Suplemento 427, 2015).

3.1.2.6. Historia clínica electrónica.

La Historia Clínica Electrónica (HCE) o también llamada Historia Clínica Digital (HCD):

Es una recopilación computarizada de los detalles de salud de un paciente. Pero es más que eso, es una nueva manera de almacenar y organizar la información del paciente. Al igual que las fichas de hospital, los archivos de EHR de los pacientes se dividen en secciones donde los profesionales entran la información para proporcionarle cuidado médico al paciente o realizar tareas administrativas (The Nemours Foundation, 2014).

La HCE es un software que nos permite crear, guardar, organizar y editar información clínica de un paciente en una computadora o en la nube. Pero es mucho más que el equivalente electrónico del papel. Están dedicadas a mejorar la eficiencia, calidad y seguridad en el cuidado de la salud.

A nivel mundial las HCE han sido acogidas con buenos frutos ya que ha demostrado beneficios que incluyen la de reducir las falencias en medicina, mejoras a nivel de costo-efectividad, aumento de la eficacia y la posibilidad de ofrecer un rol activo a los pacientes en la toma de decisiones clínicas. Que es el objetivo de cualquier sistema de información en salud.

3.1.2.7. Ficha familiar.

La ficha familiar, es una herramienta para registro de información, programación y evaluación de actividades extramurales, que realizan los EAIS durante las visitas domiciliarias. Es un instrumento que recoge la información vital de todos los habitantes de la vivienda (HOGAR) y luego permite consolidar información de las familias, que complementada con la información de la condición de salud individualizada de los usuarios, permite hacer un plan de acciones intersectoriales que mejoren la condición de vida.

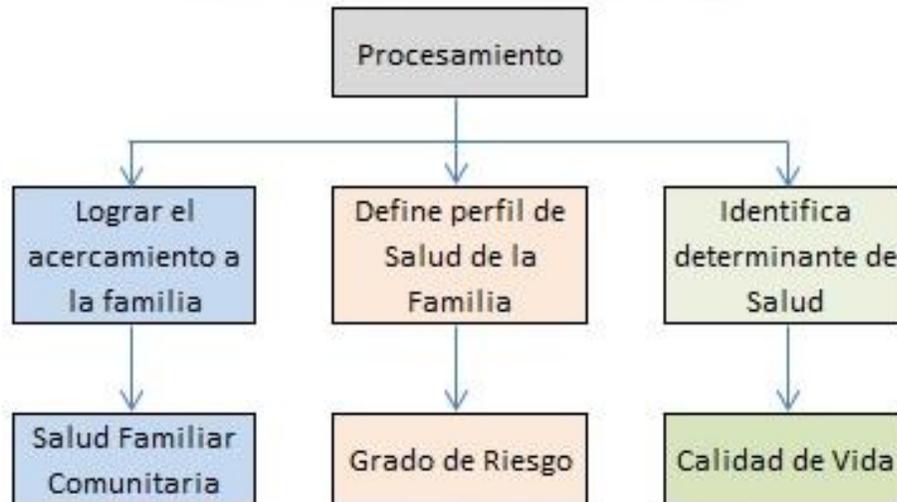
La información de las condiciones de salud de la ficha familiar, es complementaria a la información de la historia clínica individual. La Ficha Familiar es un documento que debe ser usado tanto en la actividad extramural de visitas domiciliarias, como también en el establecimiento de salud para la programación de los tiempos laborales de los EAIS, diferenciando así el tiempo para las atenciones intramurales de consulta externa general, odontología, psicología laboratorio clínico y radioterapia y más tiempo laboral que requiere para dedicarlo a las coordinaciones interinstitucionales requeridas.

- Objetivo general de la aplicación de la ficha familiar.

- Disponer de un registro integral de la información necesaria para identificar los riesgos biológicos, higiénicos sanitarios y socio-económicos de las familias del área de cobertura y aplicar el modelo de atención integral en la evaluación del riesgo familiar y la ejecución de los planes de intervención

- Implementar un plan de intervención integral para los miembros de la familia, y el respectivo seguimiento y evaluación de efectividad.

Gráfico N° 1 – Uso de información de ficha familiar



Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de investigación

3.1.2.8. Aplicación web.

Las aplicaciones web se basan en el modelo cliente/servidor. Con una diferencia importante, las conexiones no son persistentes. Es decir, en la interfaz cliente, el navegador web realiza una petición al servidor, se establece la comunicación iniciándose un diálogo entre servidor y cliente. El servidor atiende la petición y le envía el otro archivo que necesita el navegador. Una vez que es atendido el cliente, ya no es necesario mantener ninguna comunicación entre el servidor y el cliente (Tallado San Miguel, 2015, pág. 75).

- Funcionamiento de una aplicación web.

El siguiente concepto explica de qué está compuesta una aplicación web:

Una aplicación Web es un conjunto de páginas Web estáticas y dinámicas. Una página Web estática es aquella que no cambia cuando un usuario la solicita: el servidor Web envía la página al navegador Web solicitante sin modificarla. Por el contrario, el servidor modifica las páginas Web dinámicas antes de enviarlas al navegador solicitante. La naturaleza cambiante de este tipo de página es la que le da el nombre de dinámica (ADOBE, 2016).

Cuadro N° 4 – Ventajas e inconvenientes de tipos de aplicaciones

Tipo de Aplicación	Conexión Persistente	Tiempo Real	Entorno Gráfico	Necesita Red	Recursos Servidores	Multiplataforma
Aplicación de terminal	Sí	Sí	NO	Sí	Sí, bajos	Sí
Aplicación de escritorio	No	Sí	Sí	NO	NO	NO
Aplicación Cliente/ Servidor	Sí	Sí	Sí	Sí	Sí, bajos	NO
Aplicación Web	No	Sí	Sí	Sí	Sí, bajos	Sí

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de Investigación

En conclusión con el cuadro 4 podemos llegar a los siguientes resultados o ventajas de las aplicaciones web:

- La aplicación web necesitan menos recursos del servidor.
- No necesitan de un Software para cada aplicación web que vamos a usar.
- Es transportable e independiente de la plataforma. Es decir desde cualquier navegador de nuestro equipo podemos acceder a la aplicación. Incluso desde Smartphone.
- Es libre de licencias.
- El costo de mantenimiento es bajo.

Como desventajas podemos decir:

- No hay un control directo sobre los datos.
- Necesita tener una conexión a internet.
- Necesita de la vinculación con el servicio que proporciona la aplicación web.

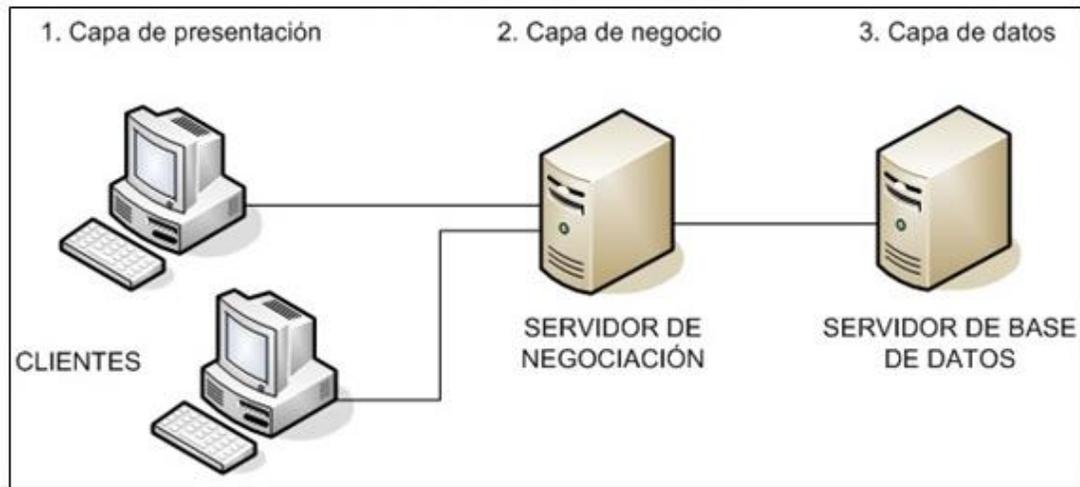
3.1.2.9. Arquitectura en n capas.

(Moquillaza Henríquez, Guerra Grados, & Vega Huerta, 2010) Afirma que:

La programación por capas es un estilo de programación en la que el objetivo principal es separar la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de negocios y ésta a su vez de la capa de presentación al usuario (p. 58).

Las capas dentro de una arquitectura son un grupo de servicios especializados que pueden ser accedidas por múltiples clientes y que pueden ser fácilmente reutilizables.

Gráfico N° 2 – Modelo típico de tres capas



(López, s.f., pág. 1)

Fuente: http://www.academia.edu/10102692/Arquitectura_de_n_capas

Tipos de capas.

-Capa de presentación.

“Es la que se presenta al usuario, llamada también formulario o interfaz de presentación, esta captura los datos del usuario en el formulario e invoca a la capa de negocio, transmitiéndole los requerimientos del usuario, ya sea de almacenaje, edición, o de recuperación de la información para la consulta respectiva” (Moquillaza Henríquez, Guerra Grados, & Vega Huerta, 2010).

-Capa de negocio.

“Esta capa interactúa con la capa de presentación para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al manejador de base de

datos que realice una operación de almacenamiento, edición, eliminación, consulta de datos u otra” (Moquillaza Henríquez, Guerra Grados, & Vega Huerta, 2010).

-Capa de datos.

“Es aquí donde se implementa las conexiones al servidor y la base de datos propiamente dicha, se invoca a los procedimientos almacenados los cuales reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio” (Moquillaza Henríquez, Guerra Grados, & Vega Huerta, 2010).

3.1.3. Estándares ¿por qué?

Cuando la diversidad crea ineficiencia o impide efectividad, se necesitan estándares. Tradicionalmente, las instituciones de salud fueron concebidas como unidades independientes donde los pacientes recibían atención médica en diferentes niveles (primario, secundario y terciario) sin ningún tipo de comunicación bi-direccional ni coordinación efectiva. La falta de coordinación entre los distintos escenarios y actores del sistema se traduce en ineficiencias en el cuidado. (Pazos, 2010, p.4)

3.1.3.1. Estándares de historia clínica electrónica.

Existen 3 estándares principales sobre HCE que son: HL7, open ERH e ISO 13606. Cada uno de estos estándares tiene su propia definición de modelos de información y

se gestiona de forma diferente. A pesar de esto tienen la característica general de seguir una arquitectura de modelo Dual; el estándar ISO 13606 se utiliza para el desarrollo del presente proyecto de titulación.

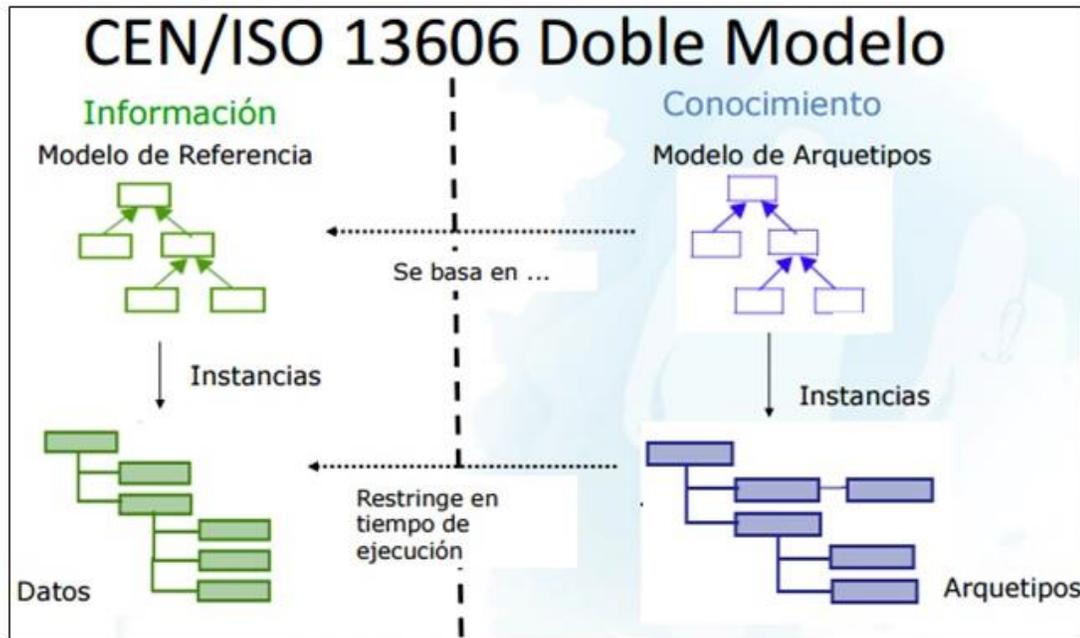
3.1.3.2. Norma CEN/ISO 13606.

Es una norma europea del Comité Europeo de Normalización (CEN) también se ha aprobado como un estándar internacional ISO. Está diseñado para conseguir interoperabilidad semántica en la comunicación electrónica del registro de salud.

El objetivo general de esta norma es definir una arquitectura de información rigurosa y estable para la comunicación de parte o la totalidad de la HCE de un solo tema de la atención que es el paciente, entre los diferentes sistemas hospitalarios. Para lograr este objetivo la norma presenta una innovadora arquitectura de doble modelo (INEN, 2008).

La arquitectura Dual Modelo define una clara separación entre información y conocimiento. El primero se estructura a través de un modelo de referencia que contiene las entidades básicas para representar cualquier información de la HCE. El segundo se basa en arquetipos, que son las descripciones formales de los conceptos clínicos.

Gráfico N° 3 – CEN/ISO 13606 doble modelado



(Muñoz Carrero, s.f., pág. 5)

Fuente:

http://www.msssi.gob.es/organizacion/sns/planCalidadSNS/docs/MUNOZ_CARRERO.pdf

La Norma fue adoptada en el Ecuador en el año 2014 y el comité nacional responsable de esta Norma Técnica Ecuatoriana y de su adopción es el Comité Interno del Instituto Ecuatoriano de Normalización (INEN, 2008).

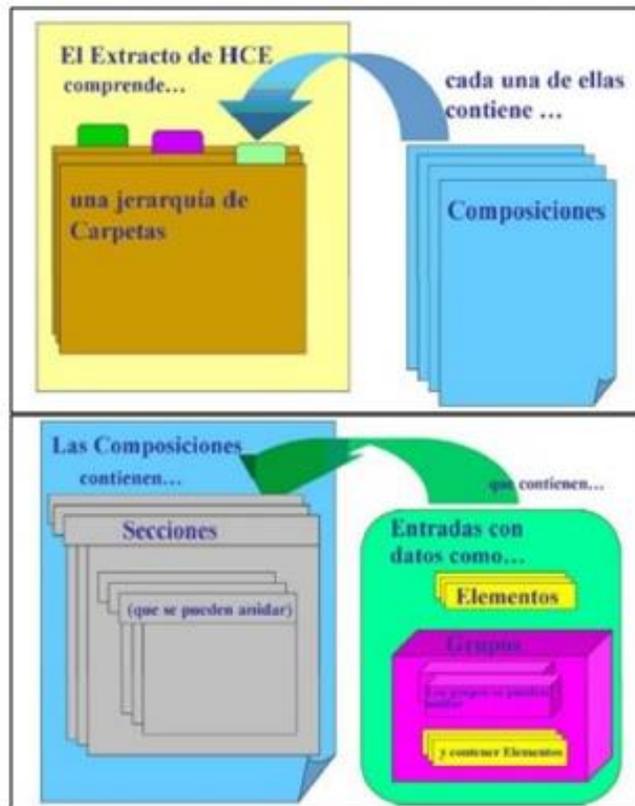
-Modelo de referencia.

Este modelo se usa para comunicar con la historia clínica electrónica de cualquier paciente.

Es su estudio (Rubio González , 2009) afirma que: “Los extractos forman parte de los mensajes, los extractos incluyen información demográfica para reconocer al paciente y a todos los agentes involucrados, información sobre las políticas de acceso,

información clínica y otros tipos de información auxiliar como auditorías o firmas.”(p.15)

Gráfico N° 4 - Estructura del modelo de referencia parte 1 y 2



(INEN, 2008, págs. 9-10)

Fuente: http://www.normalizacion.gob.ec/wp-content/uploads/downloads/2014/NORMAS_2014/DRO/nte_inen_iso_13606_1extracto.pdf

-Arquetipo.

Es la definición formal de combinaciones prescritas de las clases componentes-básicos definidas en el Modelo de Referencia para dominios u organizaciones clínicas particulares. Un arquetipo es una expresión formal de un distinto, concepto a nivel dominio, expresado en la forma de restricciones sobre los datos cuyas ocurrencias son conformes con el modelo de referencia. (INEN, 2008, pág. 8)

Gráfico N° 5 – Bloques lógicos de arquetipos



(Muñoz Carrero, s.f., pág. 9)

Fuente:

http://www.msssi.gob.es/organizacion/sns/planCalidadSNS/docs/MUNOZ_CARRERO.pdf

-Ejemplo de Arquetipo

Concepto del dominio:

Medida de presión arterial

Presión sistólica (mmHg)

Presión diastólica (mmHg)

Método:

Posición del paciente (tumbado/sentado)

Instrumento utilizado

Manguito empleado

En el gráfico 6 se muestra este ejemplo para su análisis y comprensión.

Gráfico N° 6 – Ejemplo de arquetipo

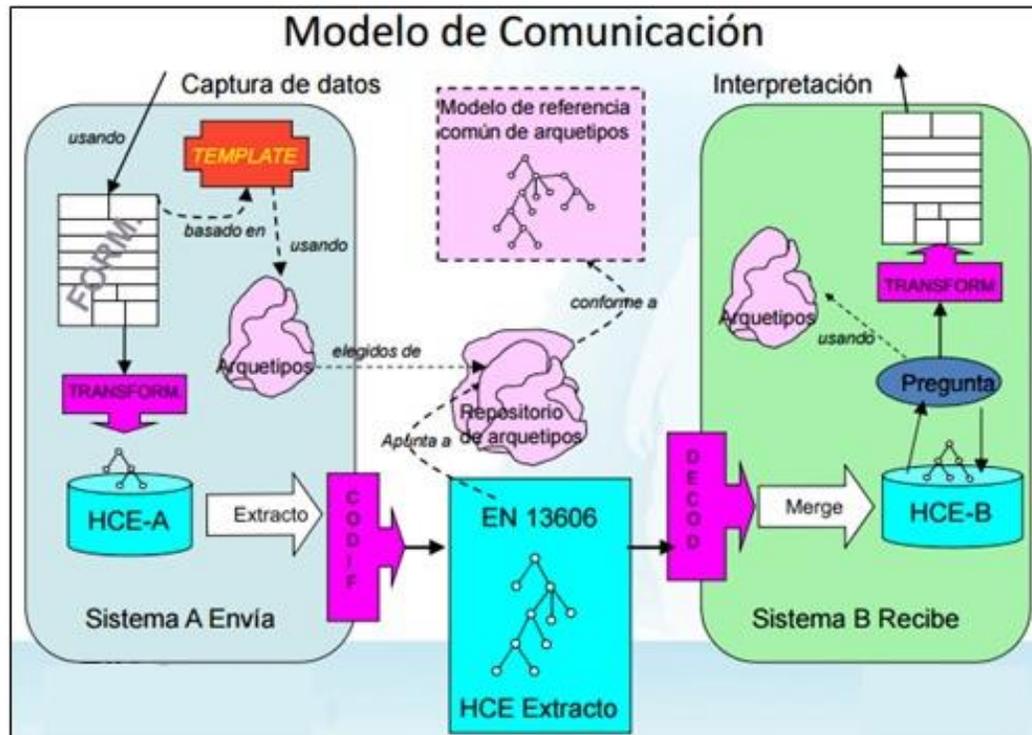


(Muñoz Carrero, s.f., pág. 12)

Fuente:

http://www.msssi.gob.es/organizacion/sns/planCalidadSNS/docs/MUNOZ_CARRERO.pdf

Gráfico N° 7 – Modelo de comunicación



(Muñoz Carrero, s.f., pág. 14)

Fuente:

http://www.msssi.gob.es/organizacion/sns/planCalidadSNS/docs/MUNOZ_CARRERO.pdf

3.1.3.3. OpenEHR.

Gráfico N° 8 – Logo OpenEHR



(Fundación OpenEHR, s. f.)

Fuente: <http://www.openehr.org/gui/Logo.jpg>

La Fundación OpenEHR es una empresa sin fines de lucro, limitada por garantía, del University College de Londres, Reino Unido. Está regulado bajo las Leyes Reino Unido de 1985 y 1989. El nombre 'openEHR' ha sido registrado internacionalmente como marca. Una de sus principales funciones legales es de titular de la Propiedad Intelectual, que se pone gratuitamente a disposición. (Fundación OpenEHR, s. f.).

El éxito de openEHR es en gran parte debido a la aceptación formal del CEN 13606 como norma europea e ISO. Esta norma se basa en muchos aspectos del enfoque de diseño openEHR, y la parte 2 de la norma es una instantánea de las especificaciones openEHR arquetipo. La Fundación openEHR trabaja en estrecha colaboración con el CEN, ISO, HL7 y organizaciones OMG y otras normas sobre estándares de modelado relacionados con la HCE y clínicos.

Cuadro N° 5 – Propiedad intelectual licencia OpenEHR

Activo	Descripción	Licencia
Especificación	Especificaciones oficiales de la Fundación	Los derechos de autor Fundación openEHR, bajo licencia Creative Commons CC-BY-ND.
Modelos Clínicos	Arquetipos, plantillas y subconjuntos, terminología elaborada por la comunidad.	Creative Commons para uso individual y organizacional. CC-BY-(SA) El compartir bajo la misma (SA) se aplica específicamente a los arquetipos derivados y plantillas únicas.
Software	El Software con derechos de propiedad intelectual adscritos a la Fundación openEHR	Licencia Apache 2

(Fundación OpenEHR, s. f.)

Fuente: http://www.openehr.org/about/intellectual_property

3.1.3.4. LinkEHR.

Gráfico N° 9 – Logo LinkEHR

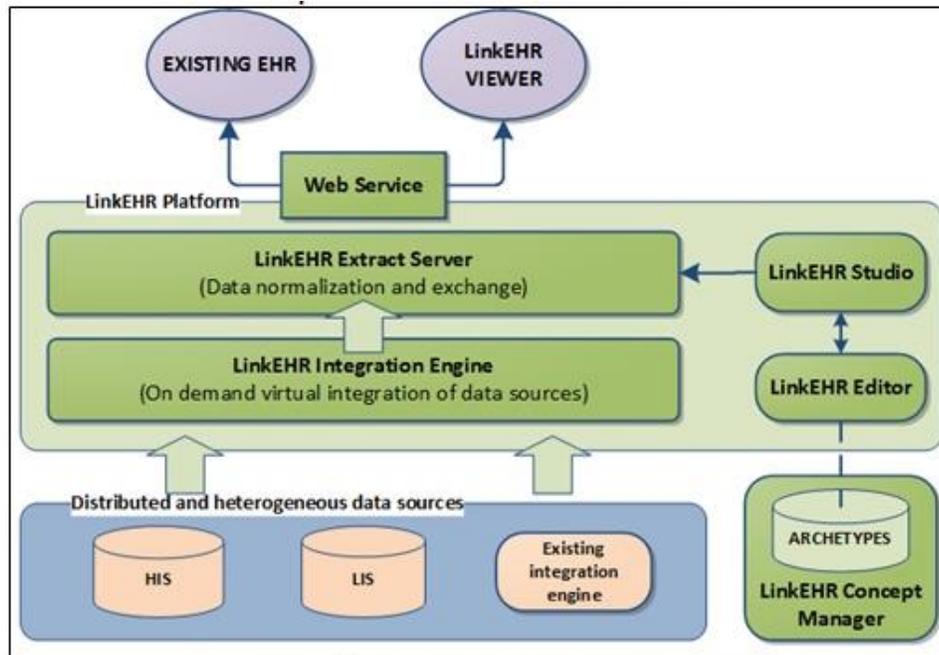


(LinkEHR, 2005-2013)

Fuente: <http://linkehr.com/>

Es una plataforma de software para el modelado, la normalización y la interoperabilidad semántica de los datos sanitarios. Con LinkEHR se puede describir con precisión los datos clínicos de una organización, transformarlo en formatos estándar como CEN/ISO 13606, HL7 CDA o cualquier otro formato, y que sea interoperable. La Plataforma LinkEHR ayuda a integrar y normalizar los datos clínicos. En esta sección se puede encontrar una breve descripción de los principales módulos de la Plataforma LinkEHR. (Plataforma LinkEHR, 2004-2015)

Gráfico N° 10 - Principales módulos de la plataforma LinkEHR



(LinkEHR, 2005-2013)

Fuente: http://pangea.upv.es/new_linkehr_web/platform.html

3.1.3.5. Herramientas de desarrollo.

Para el desarrollo de este proyecto usaremos las siguientes herramientas:

Todo el Software debe ser en 32 bits para no tener problemas de compatibilidad.

- IDE Desarrollo Eclipse Neon 32 bits "última versión"
- Capa de Presentación: JSF + Primefaces.
- Capa de Negocio: Spring
- Capa Persistencia: JPA 2.1 + Hibernate
- Base de Datos: PostgreSQL.
- JDK 1.8
- Servidor de Aplicaciones WildFly 10.0

-Java

Es un lenguaje de programación muy popular, creado en la década de los 90 del siglo XX por la empresa Sun Microsystems. Debido a la estabilidad y ventajas que ofrece hasta la actualidad muchas aplicaciones y sitios web utilizan Java para sus empresas y para enseñanza educativa. Gracias a la programación orientado objeto da la rapidez, seguridad y fiabilidad; como la tecnología se innova a pasos acelerados hoy podemos aplicar aplicaciones para dispositivos móviles, para consola de juegos (Oracle Corporation - Java, 2016).

-Eclipse

Es una plataforma de desarrollo de código abierto basada en Java. Por si misma, es simplemente un marco de trabajo y un conjunto de servicios para la construcción del entorno de desarrollo de los componentes de entrada. Afortunadamente, Eclipse tiene un conjunto de complementos, incluidas las Herramientas de Desarrollo de Java JDT (Gallardo, 2012).

La versión que utiliza el proyecto es la última, llamada Neón y puesta a disposición desde el 22 de junio del año 2016.

-Framework.

Es utilizado para normalizar los datos dentro de la organización para que la aplicación pueda ser manejada, almacenada y modificada de una forma organizada. Este da una opción de nueva funcionalidad al lenguaje de programación que estemos utilizando de una forma automática, solo realizando las correctas configuraciones. Con esto tenemos la ventaja de que podemos optimizar el código fuente dentro de nuestro desarrollo y solo utilizar las capas de las normas que se utilizan para la programación (Villalobos, Sánchez, & Gutiérrez, 2010)

En el desarrollo de software, un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. En programación, es un set de funciones o código genérico que realiza tareas comunes y frecuentes en todo tipo de aplicaciones (creación de objetos, conexión a base de datos, etc.).

-Framework JSF.

La tecnología Java Server Faces (JSF) es un marco de trabajo de interfaces de usuario del lado de servidor para aplicaciones web basadas en tecnología Java.

Los principales componentes de la tecnología Java Server Faces son:

Un API y una implementación de referencia para:

- Representar componentes de interfaz de usuario (UI) y manejar su estado.
- Manejar eventos, realizar validaciones en el lado del servidor y convertir datos.
- Definir la navegación entre páginas.
- Soportar internacionalización y accesibilidad y proporcionar extensibilidad para todas estas características.

La versión que utiliza el proyecto es la última, denominada 2.2 y puesta a disposición desde el 16 de abril del año 2013.

-Primefaces.

PrimeFaces es el estándar de facto en el desarrollo web en Java. PrimeFaces es una biblioteca ligera de componentes open source, sin configuración, y no hay dependencias requeridas. Sólo tiene que descargar PrimeFaces, añadir a la ruta de clases e importar el espacio de nombres para empezar.

La versión que utiliza el proyecto es la última, denominada 2.2 y puesta a disposición desde el 7 de junio del año 2016.

- Spring.

Spring fue estructurado para la plataforma J2EE de Java, es un framework orientado al desarrollo de aplicaciones web y su evolución ha crecido de una forma muy rápida hasta el día de hoy, donde podemos encontrar muchas variantes con el equipo de desarrolladores (Pardo, 2016)

Entre las características tenemos:

- Spring Framework es en la actualidad la referencia en el mundo de los frameworks de programación para los desarrolladores web de todo el mundo. Su éxito se fundamenta en la constante labor de investigación e innovación que realiza su equipo de desarrollo.
- Se trata de un framework que impulsa una metodología de trabajo ágil, eficiente y de buena praxis, lo que resulta en la creación de Software de elevada calidad y mantenibilidad.
- Tiene amplia compatibilidad para la integración con otros frameworks y librerías de uso común para la creación de aplicaciones web, desde Composite Views como Velocity o Tiles, APIs en capa de persistencia como Hibernate o JDO, y otra miscelánea de recursos como JavaMail, Quartz, etc.

- JPA 2.1.

Es un modelo de persistencia que está basado en un antiguo objeto plano de java, es decir de una clase u objeto sencillo que no depende de un framework. La aplicación de persistencia de java nos permite utilizar nuestra base de datos relacionales con el lenguaje de programación Java. Se las puede utilizar en las aplicaciones web y aplicaciones de clientes (Olivas, 2016).

Para ello, combina ideas y conceptos de los principales frameworks de persistencia, como Hibernate, Toplink y JDO. El mapeo objeto-relacional (es decir, la relación entre entidades Java y tablas de la base de datos, queries con nombre, etc.) se realiza mediante anotaciones en las propias clases de entidad.

Pero para entender JPA, tendremos que tener claro el concepto "persistencia". La persistencia o el almacenamiento permanente, es una de las necesidades básicas de cualquier sistema de información de cualquier tipo. En primer lugar, se propuso que el programa tratara los datos haciendo consultas directas a la base de datos. Después, se propuso trabajar con objetos, pero las bases de datos tradicionales no admiten esta opción.

Debido a esta situación, aparecieron los motores de persistencia, cuya función es traducir entre los dos formatos de datos: de registros a objetos y de objetos a registros. Persistir objetos Java en una base de datos relacional implica serializar un árbol de objetos Java en una base de datos de estructura tabular y viceversa. Esencial

es la necesidad de mapear objetos Java para optimizar velocidad y eficiencia de la base de datos.

-Unidades de persistencia.

Define un conjunto de todas las entidades (clases) que son gestionadas por la instancia del Entity Manager en una aplicación. Este conjunto de clases de entidad representa los datos contenidos en una única BBDD.

Las unidades de persistencia se definen en el fichero de configuración persistence.xml. Aquí les mostramos un ejemplo:

```
<persistence>
  <persistence-unit name="OrderManagement">
    <description>...</description>
    <jta-data-source>jdbc/MyOrderDB</jta-data-source>
    <jar-file>MyOrderApp.jar</jar-file>
    <class>com.widgets.Order</class>
    <class>com.widgets.Customer</class>
  </persistence-unit>
</persistence>
```

El presente archivo define una unidad de persistencia llamada OrderManagement Jar-file especifica los ficheros JAR en los que se encuentran las clases persistentes jta-

data-source especifica el nombre global JNDI de la fuente de datos que utiliza el contenedor.

La persistencia puede tener 4 estados diferentes:

- **Transient:** Un objeto recién creado que no ha sido enlazado con el gestor de persistencia.
- **Persistent:** Un objeto enlazado con la sesión (Todos los cambios serán persistentes).
- **Detached:** Un objeto persistente que sigue en memoria después de que termina la sesión: existe en java y en la BBDD.
- **Removed:** Un objeto marcado para ser eliminado de la BBDD: existe en java y se borrará de la BBDD al terminar la sesión.

-Hibernate.

Es una herramienta que se utiliza con el lenguaje de programación Java, su función es Mapeo objeto relacional lo que permite un mapeo de atributos entre la base de datos relacional y un modelo de objetos de la aplicación que se está desarrollando. Mediante los archivos de tipo Xml o anotaciones de las entidades que permiten establecer estas relaciones. Hibernate es software libre, distribuido bajo los términos de la licencia GNU LGPL (Educación IT, 2013).

En otras palabras, Hibernate es un Framework que agiliza la relación entre la aplicación y la base de datos. Para poder aprender a utilizarlo es necesario contar con los conocimientos básicos de base de datos y SQL así como manejar el lenguaje Java.

¿Por qué usar un Framework ORM?

Cuando desarrollamos aplicaciones en muchos casos todo termina siendo un conjunto de ABM (alta, baja y modificaciones de datos) que luego consultamos. Para ello se utiliza una base de datos donde hay muchas tareas repetidas: por cada objeto que quiero persistir debo crear una clase que me permita insertarlo, eliminarlo, modificarlo y consultarlo. Con excepción de consultas especiales, el resto es siempre lo mismo. Este es el momento dónde un ORM tiene una importancia fundamental. Con solo configurarlo todas estas tareas se ejecutan automáticamente y solo tendrás que preocuparte por las consultas especiales. (Educación IT, 2013)

¿Cómo funciona Hibernate?

El desarrollador deberá configurar en un archivo XML o mediante anotaciones donde corresponde un atributo de una clase, con una columna de una tabla. Es una tarea simple donde existen herramientas que lo hacen por nosotros. (Educación IT, 2013)

-PostgreSQL.

Es una base de datos de objetos relacional, que tiene un costo por licencia BSD y su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado (Martínez, 2009-2013).

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

Características

- Es una base de datos 100% ACID
- Integridad referencial
- Tablespaces
- Nested transactions (savepoints)
- Replicación asincrónica/sincrónica / Streaming replication - Hot Standby
- Two-phase commit
- PITR - point in time recovery
- Copias de seguridad en caliente (Online/hot backups)
- Unicode
- Juegos de caracteres internacionales
- Regionalización por columna
- Multi-Version Concurrency Control (MVCC)
- Múltiples métodos de autenticación
- Acceso encriptado via SSL
- Actualización in-situ integrada (pg_upgrade)
- SE-postgres
- Completa documentación
- Licencia BSD
- Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit.

-JDK 1.8.

Es un software que facilita las herramientas de desarrollo para la creación de aplicaciones en el lenguaje de programación java. Puede instalarse en un dispositivo local o en una unidad de red compartida (Latorre, 2010).

En los sistemas Windows sus variables de entorno son:

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

Los programas más importantes que se incluyen son:

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

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

-Servidor de aplicaciones WILDFLY 10.0.

Esta herramienta nos permite cambios más flexibles y modernos para el desarrollo de las aplicaciones empresariales en Java. Nos permite una mayor participación de la comunidad, tener la capacidad de operar en los entornos de nubes híbridas y tener así soporte tecnológico en caso de presentarse un inconveniente durante nuestro desarrollo (csi2altair, 2016).

Esta tecnología continuará sirviendo como proyecto de desarrollo preliminar de JBoss Enterprise Application Platform de Red Hat y se centra en algunas de las principales fuerzas que plasman el middleware hoy en día, incluso el cambio hacia enfoques más flexibles y modernos para el desarrollo de aplicaciones, la habilitación de nubes híbridas abiertas y Java Enterprise Edition 7 (Java EE 7). “WildFly continúa la tradición de una década de Red Hat JBoss Middleware de desafiar los límites del desarrollo de software empresarial” expresó el Dr. Mark Little, Vicepresidente de Ingeniería de Middleware de Red Hat. “Además, también representa una clara oportunidad para una mayor adopción y participación por parte de la comunidad“.

3.1.4. Fundamentación legal.

Dentro de la constitución existen leyes que respaldan la elaboración de este proyecto:

3.1.4.1. Ley orgánica de salud.

Art. 7.- Toda persona, sin discriminación por motivo alguno, tiene en relación a la salud, los siguientes derechos (...) f) Tener una historia clínica única redactada en términos precisos, comprensibles y completos; a Sí como la confidencialidad respecto de la información en ella contenida y a que se le entregue su epicrisis. (Congreso Nacional, 2012, pág. 5)

3.1.4.2. Ley orgánica del sistema nacional de salud.

Art. 11.-...El Sistema establecerá los mecanismos para que las instituciones garanticen su operación en redes y aseguren la calidad, continuidad y complementariedad de la atención, para el cumplimiento del Plan Integral de Salud de acuerdo con las demandas y necesidades de la comunidad. (Congreso Nacional, 2012, pág. 6)

Estos mecanismos incluyen:...

d) “Un conjunto común de datos básicos para la historia clínica;

Art. 27.-El Ministerio de Salud Pública, con el apoyo del Consejo Nacional de Salud, implantará y mantendrá un sistema común de información sectorial que permitirá conocer la situación de salud, identificar los riesgos para la salud de las personas y el

ambiente, dimensionar los recursos disponibles y la producción de los servicios para orientar las decisiones políticas y gerenciales en todos los niveles. (Congreso Nacional, 2012, pág. 11)

3.1.4.3. Ley de derechos y amparo al paciente.

Art. 4.- DERECHO A LA CONFIDENCIALIDAD.- Todo paciente tiene derecho a que la consulta, examen, diagnóstico, discusión, tratamiento y cualquier tipo de información relacionada con el procedimiento médico a aplicársele, tenga el carácter de confidencial. (Congreso Nacional, 2006, pág. 2)

3.1.4.4. Reglamento de información confidencial en sistema nacional de salud.

Art. 6.- Secreto Médico.- Es la categoría que se asigna a toda información que es revelada por un/a usuario/a al profesional de la salud que le brinda la atención de salud. Se configura como un compromiso que adquiere el médico ante el/la usuario/a y la sociedad, de guardar silencio sobre toda información que llegue a conocer sobre el/la usuario/a en el curso de su actuación profesional. (Ministro de Salud Pública, 2015, pág. 3)

Art. 14.- La historia clínica sólo podrá ser manejado por personal de la cadena sanitaria. Como tal se entenderá a los siguientes profesionales: médicos, psicólogos, odontólogos, trabajadoras sociales, obstetricas, enfermeras, además de auxiliares de enfermería y personal de estadística. (Ministro de Salud Pública, 2015, pág. 4)

Art. 15.- El acceso a documentos archivados electrónicamente será restringido a personas autorizadas por el responsable del servicio o del establecimiento, mediante claves de acceso personales. (Ministro de Salud Pública, 2015, pág. 4)

Art. 16.- La custodia física de la historia clínica es responsabilidad de la institución en la que repose. El personal de la cadena sanitaria, mientras se brinda la prestación, es responsable de la custodia y del buen uso que se dé a la misma, generando las condiciones adecuadas para el efecto. (Ministro de Salud Pública, 2015, pág. 4)

Art. 17.- El archivo de historias clínicas es un área restringida, con acceso limitado solo a personal de salud autorizado, donde se guardan de manera ordenada, accesible y centralizada todas las historias clínicas que se manejan en el establecimiento. Se denomina activo cuando cuenta con historias activas, esto es con registros de hasta cinco años atrás y se denomina pasivo cuando almacena aquellas que tienen más de cinco años sin registros, tomando en cuenta la última atención al paciente. (Ministro de Salud Pública, 2015, pág. 4)

Art. 18.- Los datos y la información consignados en la historia clínica y los resultados de pruebas de laboratorio e imagenología registrados sobre cualquier medio de soporte ya sea físico, electrónico, magnético o digital, son de uso restringido y se manejarán bajo la responsabilidad del personal operativo y administrativo del establecimiento de salud, en condiciones de seguridad y confidencialidad que impidan que personas ajenas puedan tener acceso a ellos. (Ministro de Salud Pública, 2015, pág. 4)

Art. 19.- Todas las dependencias que manejen información que contenga datos relevantes sobre la salud de los/las usuarios/as deberán contar con sistemas adecuados de seguridad y custodia. (Ministro de Salud Pública, 2015, pág. 4)

Art. 20.- Los documentos físicos que contengan información confidencial de los/las usuarios/as y que no requieran ser archivados, deberán ser destruidos evitando su reutilización, de conformidad a lo dispuesto en el Capítulo II del Manual del Manejo, Archivo de las Historias Clínicas. (Ministro de Salud Pública, 2015, pág. 4)

3.1.4.5. Acuerdo del ministerio de salud pública No. 0138.

Art. 1.- Aprobar y publicar los formularios básicos actualizados de la historia clínica única de acuerdo a la numeración y nomenclatura establecida. (Ministra de Salud Pública, 2008, pág. 13)

Art. 2.- Los números y nombres de los formularios básicos de la historia clínica única serán los siguientes:

001 ADMISION Y ALTA

002 CONSULTA EXTERNA ANAMNESIS Y EXAMEN FISICO

003 ANAMNESIS Y EXAMEN FISICO

005 EVOLUCION Y PRESCRIPCIONES

006 EPICRISIS

007 INTERCONSULTA

008 EMERGENCIA

010 LABORATORIO CLINICO

012 IMAGENOLOGIA

013 HISTOPATOLOGIA

020 SIGNOS VITALES

021 ADMINISTRACION DE MEDICAMENTOS

024 AUTORIDADES Y CONSENTIMIENTO INFORMADO
033 ODONTOLOGIA
038 TRABAJO SOCIAL
053 REFERENCIA Y CONTRAREFERENCIA
054 CONCENTRADO DE LABORATORIO
055 CONCENTRADO DE LABORATORIO EXÁMENES ESPECIALES

(Ministra de Salud Pública, 2008, pág. 13)

Art. 3.- Serán parte complementaria de la Historia Clínica Única los siguientes instrumentos:

ANEXO – 1 FICHA FAMILIAR
ANEXO – 2 ATENCIÓN PRE – HOSPITALARIA
MANUAL DE USO

(Ministra de Salud Pública, 2008, pág. 13)

Art. 4.- Disponer a todos los establecimientos de salud públicos y privados a nivel nacional la inmediata aplicación de los formularios básicos actualizados y anexos.

(Ministra de Salud Pública, 2008, pág. 13)

Art. 5.- El consejo Nacional de Salud (CONASA) concertará con todas las entidades del Sistema Nacional de Salud, la difusión y aplicación obligatoria de los formularios y anexos señalados en el presente Acuerdo. (Ministra de Salud Pública, 2008, pág. 13)

3.1.4.6. Reglamento a la ley orgánica del sistema nacional de salud.

(Presidente Noboa Bejarano, 2003) En su Decreto Ejecutivo N° 3611 dispone en el capítulo IV:

Art. 76.- De la clasificación internacional de enfermedades.- La aplicación y uso de la clasificación internacional de enfermedades será obligatoria en todos los niveles y entidades del Sistema Nacional de Salud.

Art. 77.- De la Historia Clínica Única.- El Ministerio de Salud Pública, en su calidad de autoridad sanitaria, revisará y actualizará los formularios básicos y de especialidades de la historia clínica única para la atención de los usuarios, los mismos que serán posteriormente concertados y difundidos por el Consejo Nacional de Salud en todas las instituciones prestadoras de salud del sector público, semipúblico y privado.

Art. 78.- Obligatoriedad de uso de la historia clínica única.- El uso y aplicación de la historia clínica única serán obligatorios en las instituciones de salud que integran el sistema.

Art. 79.- Responsabilidad y uso de la historia clínica.- La historia clínica, en tanto prueba documental, estará bajo la responsabilidad y custodia de la unidad o establecimiento operativo del lugar de residencia del ciudadano; se propenderá que exista una sola historia clínica por persona, que será la base para el sistema de referencia y contra referencia. (pág. 25)

Disposición transitoria 4.- El plazo para el diseño de los formularios básicos que forman parte de la historia clínica única será de un año calendario, a partir de la publicación en el Registro Oficial del presente Reglamento.

3.1.4.7. Ley orgánica de educación superior.

Art. 8.- Serán Fines de la Educación Superior.-

La educación superior tendrá los siguientes fines:

- a) Aportar al desarrollo del pensamiento universal, al despliegue de la producción científica y a la promoción de las transferencias e innovaciones tecnológicas:
- b) Contribuir al conocimiento. Preservación y enriquecimiento de los saberes ancestrales y de la cultura nacional.

(Presidente Correa Delgado, 2010, pág. 6)

Art. 32.- Programas informáticos.- Las empresas que distribuyan programas informáticos tienen la obligación de conceder tarifas preferenciales para el uso de las licencias obligatorias de los respectivos programas, a favor de las instituciones de educación superior, para fines académicos. Las instituciones de educación superior obligatoriamente incorporarán el uso de programas informáticos con software libre.

(Presidente Correa Delgado, 2010, pág. 10)

3.1.4.8. Principios del sistema de educación superior.

Art. 13.- Funciones del Sistema de Educación Superior.- Son funciones del Sistema de Educación Superior:

- a) Garantizar el derecho a la educación superior mediante la docencia, la investigación y su vinculación con la sociedad, y asegurar crecientes niveles de calidad, excelencia académica y pertinencia.
- b) Promover la creación, desarrollo, transmisión y difusión de la ciencia, la técnica, la tecnología y la cultura.
- c) Formar académicos, científicos y profesionales responsables, éticos y solidarios, comprometidos con la sociedad, debidamente preparados para que sean capaces de generar y aplicar sus conocimientos y métodos científicos, así como la creación y promoción cultural y artística.

3.1.4.9. Ley de la propiedad intelectual.

El (Congreso Nacional, 1998) en el capítulo V dispone:

Art. 28. Los programas de ordenador se consideran obras literarias y se protegen como tales. Dicha protección se otorga independientemente de que hayan sido incorporados en un ordenador y cualquiera sea la forma en que estén expresados, ya sea en forma legible por el hombre (código fuente) o en forma legible por máquina (código objeto), ya sean programas operativos y programas aplicativos, incluyendo

diagramas de flujo, planos, manuales de uso, y en general, aquellos elementos que conformen la estructura, secuencia y organización del programa.

Art. 29. Es titular de un programa de ordenador, el productor, esto es la persona natural o jurídica que toma la iniciativa y responsabilidad de la realización de la obra. Se considerará titular, salvo prueba en contrario, a la persona cuyo nombre conste en la obra o sus copias de la forma usual.

Dicho titular está además legitimado para ejercer en nombre propio los derechos morales sobre la obra, incluyendo la facultad para decidir sobre su divulgación.

El productor tendrá el derecho exclusivo de realizar, autorizar o prohibir la realización de modificaciones o versiones sucesivas del programa, y de programas derivados del mismo.

Las disposiciones del presente artículo podrán ser modificadas mediante acuerdo entre los autores y el productor.

Art. 30. La adquisición de un ejemplar de un programa de ordenador que haya circulado lícitamente, autoriza a su propietario a realizar exclusivamente:

- a) Una copia de la versión del programa legible por máquina (código objeto) con fines de seguridad o resguardo;
- b) Fijar el programa en la memoria interna del aparato, ya sea que dicha fijación desaparezca o no al apagarlo, con el único fin y en la medida necesaria para utilizar el programa;
- c) Salvo prohibición expresa, adaptar el programa para su exclusivo uso personal, siempre que se limite al uso normal previsto en la

licencia. El adquirente no podrá transferir a ningún título el soporte que contenga el programa así adaptado, ni podrá utilizarlo de ninguna otra forma sin autorización expresa, según las reglas generales.

Se requerirá de autorización del titular de los derechos para cualquier otra utilización, inclusive la reproducción para fines de uso personal o el aprovechamiento del programa por varias personas, a través de redes u otros sistemas análogos, conocidos o por conocerse.

Art. 31. No se considerará que exista arrendamiento de un programa de ordenador cuando éste no sea el objeto esencial de dicho contrato. Se considerará que el programa es el objeto esencial cuando la funcionalidad del objeto materia del contrato, dependa directamente del programa de ordenador suministrado con dicho objeto; como cuando se arrienda un ordenador con programas de ordenador instalados previamente. (pág. 13)

3.1.4.10. Sobre el uso de software libre.

La constitución de la República del Ecuador a través del Congreso Nacional (2008) decreto la ley de la propiedad intelectual que menciona:

Art. 1. Establecer como política pública para las entidades de administración pública central la utilización del Software Libre en sus sistemas y equipamientos informáticos.

Art. 2. Se entiende por Software Libre, a los programas de computación que se pueden utilizar y distribuir sin restricción alguna, que permitan el acceso a los códigos fuentes y que sus aplicaciones puedan ser mejoradas.

Estos programas de computación tienen las siguientes libertades:

- a) Utilización de programa con cualquier propósito de uso común.
- b) Distribución de copias sin restricción alguna.
- c) Estudio y modificación de programa (Requisito: código fuente disponible)
- d) Publicación del programa mejorado (Requisito: código fuente disponible).

Art. 3. Las entidades de la administración pública central previa a la instalación del software libre en sus equipos, deberán verificar la existencia de capacidad técnica que brinde el soporte necesario para este tipo de software.

Art. 4. Se faculta la utilización de software propietario (no libre) únicamente cuando no exista una solución de Software Libre que supla las necesidades requeridas, o cuando esté en riesgo de seguridad nacional, o cuando el proyecto informático se encuentre en un punto de no retorno.

3.1.5. Pregunta científica a contestarse.

¿La implementación del formulario Ficha Familiar dentro del programa e-Salud, mejorará el proceso de atención médica cumpliendo con las demandas y necesidades de la comunidad?

3.1.6. Definiciones conceptuales.

Aplicación Web: herramienta que permite a los usuarios, acceder a un servicio web a través del internet, esto quiere decir que todos los datos que trabajamos son procesados y almacenados dentro de la web. Por éste motivo éste concepto se encuentra estrechamente relacionado con la nube.

ISO 13606: Especifica la comunicación de parte o la totalidad de la HCE de un solo sujeto identificado de la atención sistemas historia clínica electrónica (ISO, 2008).

Interoperabilidad: Procesos, tecnologías y protocolos requeridos para asegurar la integridad de los datos cuando se transfieren de un sistema a otro.

4. Capítulo III

4.1. Propuesta tecnológica

La propuesta del presente proyecto es el desarrollo del formulario de anexo 1 - Ficha familiar, aplicando Arquetipos según la norma ISO 13606, el mismo que forma parte de los formularios básicos aprobados por el MSP.

El desarrollo de éste formulario y su posterior uso, busca obtener una óptima atención a las distintas familias por parte del EAIS, mediante la automatización de los procesos de ingresos y consultas, beneficiando a los profesionales de salud otorgándoles una herramienta tecnológica para mejorar la calidad de atención para con el paciente y poder implementar un plan de intervención integral para los miembros de las familias de los diferentes sectores del país.

Este formulario se desarrolla en un ambiente web, al que se puede acceder desde cualquier ordenador que disponga una conexión a internet, y está disponible para los usuarios a través de cualquier navegador web.

4.2. Análisis de factibilidad

Se realiza el análisis de factibilidad del proyecto, es decir, se verifica si el proyecto que se desarrolla cumple con los lineamientos solicitados por la organización.

En la factibilidad operacional describimos la aceptación del producto por parte de los Usuarios en nuestro caso será los médicos y el personal que el Ministerio de Salud Pública disponga para poder realizar las entrevistas del formulario de Ficha Familiar.

En la factibilidad técnica se hace referencia a la calidad del software que creamos, los métodos que implementamos al momento de programar y los conocimientos adquiridos en la Universidad que se empleó en la investigación y desarrollo del mismo.

En la factibilidad legal se demostrará que la aplicación web que se desarrolló no incumple ninguna violación de ley de propiedad y normas que se estipulan en el software informático.

En la factibilidad económica se determina si nuestra solución genera beneficios mayores a los costos para el Ministerio de Salud Pública.

4.2.1. Factibilidad operacional.

En las entrevistas que se realizaron con los médicos de diferentes subcentros de salud del Ministerio de Salud Pública, se pudo observar el apoyo de los mismos y la aceptación del producto de software como una innovación tecnológica y una herramienta de ayuda en su trabajo del día a día con respecto a la recolección de información de las fichas familiares en diferentes puntos del país.

Como el proceso que tienen actualmente se genera de forma manual nos indicaron las diferentes dificultades que pasan con la recolección de información, así como también al momento de archivar las fichas familiares indican que se van manchando de tinta de bolígrafo y deteriorando el papel según su tiempo de vida útil.

Por lo que en muchas ocasiones se ha perdido información ocasionando inconvenientes con los Supervisores del Programa de Equipos Básicos de Salud del Ministerio de Salud Pública.

Nos dieron su opinión de apoyo en tener un lugar donde la información no se puede perder y la comunidad de un determinado subcentro u hospital no se vea afectado por estos temas, cuando ya existe la tecnología donde nos podemos comunicar y guardar información para poder reutilizarla para un beneficio de bien común para la sociedad.

EL apoyo se ha recibido por parte de los médicos de la Facultad de Medicina que comprenden el Programa Promeinfo y los médicos que se han entrevistado en los diferentes subcentros de salud del MSP.

4.2.2. Factibilidad técnica.

Para la elaboración del producto de software e-Salud del programa Promeinfo que tiene el apoyo de la Universidad de Guayaquil y el Ministerio de Salud Pública, se tiene una infraestructura que contemplan herramientas que brindan la calidad a la aplicación, a su vez con la arquitectura Modelo – Vista – Controlador (MVC) podemos dar un orden en la lógica de programación siguiendo con los estándares que está sujeto el desarrollo del software y sobre todo poder realizar la reutilización del código fuente lo que hace más liviana nuestra aplicación web.

Con los métodos de programación que se aplican podemos realizar nuestros escenarios de pruebas internas de calidad, así como asegurar que si el médico requiere un cambio luego de salir el producto a Producción este será fácil de manejar y poder ser entendido por cualquier desarrollador de software que en esos momentos este dando el soporte a la aplicación e-Salud.

A continuación se detallan las herramientas de software a utilizar y los requerimientos mínimos de hardware que se requieren para el desarrollo del proyecto:

Cuadro N° 6 – Características de hardware

Equipo de Desarrollo	Características Técnicas
PC de desarrollo	Procesador: Intel Core i3 4ta generación
	Memoria RAM: 4 Gb.
	Disco Duro: 500 Gb.
	Sistema Operativo: Windows 8 de 32 bits
Servidor	Procesador: Intel Xeon
	Memoria RAM: 8 Gb.
	Disco Duro: 2 Tb.
	Sistema Operativo: Windows Server

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de investigación

Cuadro N° 7 – Herramientas de software implementadas

Tipo de Software	Nombre
Lenguaje de Programación:	Java
IDE de Desarrollo:	Eclipse Neon
Servidor de Desarrollo Web:	Wildfly 10.0
Framework de Desarrollo:	JSF + Primeface
Framework de Persistencia:	JPA 2.1 + Hibernate
Gestor de Base de Datos:	PostgreSQL
Capa de Negocio:	Spring

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de investigación

4.2.3. Factibilidad legal.

El software e-Salud no infringe de ninguna forma posible lo establecido según las normas, estatus y reglamentos de la ley de propiedad intelectual y de software, ya que se está usando herramientas de libre acceso y código fuente abierto para no tener la necesidad de adquirir alguna licencia, como se estableció en el decreto N° 1014 Art. 1 dispuesto por el Presidente de la República que garantiza el completo y absoluto derecho al uso de software libre en las instituciones públicas del Ecuador estando cumpliendo así lo establecido por la ley.

Con Acuerdo Ministerial 01034 de 01 de noviembre de 2011, publicado en el Registro Oficial No. 279 de 30 de marzo 2012, se emite el Estatuto Orgánico de Gestión Organizacional por Procesos del MSP, mismo que dispone como misión de la Dirección Nacional de Tecnologías de la Información y Comunicaciones: “Proponer, implementar y administrar políticas, normas y procedimientos que optimicen la gestión y administración de las tecnologías de la información y comunicaciones (TIC’s), garantizando la integridad de la información, optimización de recursos, sistematización y automatización de los procesos institucionales, así como el soporte tecnológico institucional”(Registro Oficial N°. 279, 2012, p.99).

De la misma manera se indica que no se está faltando al derecho de autor ya que todas las ideas y conocimientos tomados de artículos científicos, libros y revistas han sido debidamente citados y referenciados como lo establece las normas APA en sus 6ta edición.

4.2.4. Factibilidad económica.

Para la implementación del software e-Salud hemos utilizado herramientas tecnológicas de código fuente abierta (Open Source), es decir de libre acceso permitiendo que el desarrollo de nuestra aplicación sea factible con respecto a los costos que se tienen que invertir por un software que tenga herramientas con licencia.

Otro punto a considerar es que no se necesita invertir en costos de capacitaciones sobre la aplicación web e-Salud ya que los usuarios son personas que frecuentemente usan el internet y en más de una ocasión han llenado un formulario médico.

Los beneficios son contar con una aplicación web que se pueda comunicar entre distintos puntos del Ecuador, un desafío impensable en lo que respecta con el Área de la Salud dentro del país. Además que se puede automatizar procesos como ingreso de información de la población que puede servir para el Instituto Nacional de Economía y Censos ahorrando gastos para el país, con ese dinero se puede realizar obras sociales que necesita la comunidad del Ecuador.

El presupuesto con el cual se trabajó para la realización del formulario es el siguiente:

Cuadro N° 8 – Costo de proyecto estudiante

CONCEPTO	VALOR (Dólares)	TOTAL (Dólares)
HERRAMIENTAS DE PROYECTO		\$1330.00
Computadora de escritorio	600.00	
Laptop	700.00	
Módem	30.00	
RECURSO HUMANO		\$3700.00
2 desarrolladores de software (2 meses)	3200.00	
1 Asesoría de experto	500.00	
SERVICIOS VARIOS		\$442.00
Impresiones y empastado de manual de usuario.	50.00	
Traslados	120.00	
Alimentación	240.00	
Planilla de luz	32.00	
TOTAL		\$5472.00

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de investigación

4.3. Metodología del proyecto y sus etapas

En el primer capítulo se indicó de una forma macro sobre las metodologías ágiles, de estas la implementada en este proyecto es la SCRUM.

SCRUM:

Raya (2014) afirma: “Es un marco de trabajo que nos proporciona una serie de herramientas y roles para, de una forma iterativa, poder ver el progreso y los resultados de un proyecto” (p. 3).

4.3.1. Etapas de la Metodología del Proyecto.

- Se realiza una reunión con las personas relacionadas con la problemática (médicos) para así obtener información real y confiable para elaborar una solución factible.
- La recopilación de la información es realizada en los establecimientos del MSP para de esa forma establecer en el mismo campo en que se desenvuelve los circunstantes, por otra parte se requiere de información adicional en libros o internet y revisar si existen referencias que sirvan de base para la metodología.
- El equipo de trabajo en el actual proyecto está formado especialmente por desarrolladores postulante a ingeniero de sistemas y con el siempre respaldo de las personas previamente descritas abordando la temática cada vez que sea necesario, surgiendo las preguntas que pueden proporcionar soluciones y por consiguiente la hipótesis.
- Se ejecuta la revisión reflexiva de cada una de las fuentes de información previamente obtenida, extrayendo así de los paradigmas.
- Los parámetros que delimitan el desarrollo del formulario se obtienen de los paradigmas establecidos del análisis.
- Se especifican las tecnologías más adecuadas para el proceso.

4.4. Análisis de requerimientos

4.4.1. Objetivos de la fase.

Esta fase tiene como propósito establecer los aspectos que debe cubrir el sistema, mediante el levantamiento de información del Formulario de Ficha Familiar dispuestos por el MSP para definir las especificaciones de lo que debe cumplir la aplicación web, considerando que no existe la posibilidad de solicitar nuevos resultados, una vez finalizada esta etapa.

4.4.2. Técnicas Utilizadas.

4.4.2.1. Entrevistas.

Se realizó entrevista con el Dr. Ronald Alvarado, para obtener la información de necesidades de registro de información en el Formulario Anexo1-Ficha Familiar que se encuentra en el acuerdo Ministerial N° 138, luego de la entrevista nos hizo llegar el formato impreso de la ficha familiar que está compuesta de 8 hojas, se realizó la distribución de las mismas entre los dos grupos que tratamos este formulario, en los anexos 3, 4, 5 y 6 encontramos las hojas 1, 3, 6 y 8 respectivamente, que serán las desarrolladas y validadas en este proyecto.

Para evitar lo que se denomina “conflicto de intereses” (debido a que el Dr. Alvarado forma parte de la dirección del proyecto), recurrimos a otra fuente que está aprobada por el MSP sobre el proyecto e-Salud, el Dr. Eloy Rivera que nos permitió recabar la siguiente información:

- Sobre el llenado actual de la ficha

El médico encargado de recabar la información, acude al domicilio de la familia con la ficha impresa (desde un archivo en Excel), el llenado lo realiza de forma manual y siendo este formato físico es posible que se deteriore con el pasar del tiempo o manipulación del mismo.

- Sobre el llenado deseado en la ficha mediante el sistema e-Salud

Como se trata de automatizar este proceso que como se indicó, es manual, no solo se necesita tener una aplicación que me permita registrar en la base de datos, sino también permitir que información que puede ser común en varias fichas sea seleccionada en el sistema podría ser mediante el uso de barras desplegables con las diferentes opciones de la información deseada, obviamente esta información debería estar dentro de una base de datos para ser extraída.

- Sobre el flujo de los formularios

Los formularios dependen de información de otros formularios, aunque Ficha Familiar sea un formulario anexo, la información es requerida por formularios como el 002-Consulta Externa y el 038 Trabajo Social, puede o debe existir retroalimentación entre estos formularios por lo cual es necesario que revisen los puntos compatibles entre estos.

- Sobre el orden de las preguntas en la Ficha Familiar

La información debe ser llenada siguiendo el orden del formato impreso que es el aprobado por el MSP, deben realizar las validaciones correspondientes ya que en unos casos las preguntas dependerán de otras para ser contestadas o no, por ejemplo en la hoja 3 si está llenando la ficha de un hombre de 12 años en frente de la pregunta 20 debe automáticamente saltar a la pregunta 22, así mismo en caso de que

se estuviera llenando la ficha de una mujer mayor de 12 años no llena la pregunta 18 y en la 20 si la respuesta es 0 debe automáticamente pasar a la pregunta 22.

- Sobre un módulo de búsqueda, si es necesario ¿Cuál debería ser el criterio de búsqueda?

Por supuesto que es necesario un módulo de búsqueda, aunque en los centros médicos manejan su propio criterio de búsqueda entre los cuales tienen: el número de ficha, número de la HC, número de carpeta, etc. El mejor criterio de búsqueda debe ser el número de cédula ya que es una información única para cada paciente.

- En el formato provisto por el Dr. Alvarado en la hoja 1 – Sección II existen 6 respuestas pero solo 4 ítems ¿Cuáles serían los 2 ítems faltantes? ¿Serán solo 4 respuestas en total?

Efectivamente se nota que hay un error en el formato impreso, pero el MSP hizo llegar un instructivo en formato PDF en el cual indica la forma correcta de llenar el formulario, para su pregunta son 6 ítems en total y los 2 faltantes son: 3. En construcción y 4. Destruída

Aclaraciones en la entrevista:

Se indicó que el diseño de las pantallas será secuencial, es decir, una vez que se ha llenado cierta información se procederá a dar clic en botón siguiente, para poder avanzar a las siguientes preguntas. Se tomará en cuenta la observación del Dr. Eloy en cuanto a las validaciones en preguntas recursivas.

Se explicó que se estableció un estándar en el diseño de las pantallas (interfaz de usuario) en cuanto a colores y logo que serán usados en todos los formularios del proyecto e-Salud.

Conclusiones de la Entrevista:

En las entrevistas realizadas pudimos aclarar ciertas dudas que surgieron durante el análisis y desarrollo del formulario, se revisaron ciertas observaciones propuesta por el Dr. Rivera, se hicieron ciertas aclaraciones con respecto al formato estándar de pantallas o interfaz de usuario que llevarán en general todos los formularios del proyecto e-Salud, luego de la entrevista el Dr. Rivera nos hizo llegar el Instructivo para el equipo de atención integral en salud en el manejo de la ficha familiar, está como anexo 11 en el presente documento.

4.4.2.2. Reuniones.

Se realizaron reuniones con el Dr. Ronald Alvarado y el Ing. Jorge Medina, para que se brindara información de cómo se lleva a cabo el proceso desde el registro de los pacientes, para conocer sobre el formulario implementado y además sobre las tablas que se encuentran en la base de datos para almacenar los registros.

4.4.2.3. Revisión de Registros.

Como participación en el análisis de la información se revisaron los formatos de Excel que se utilizan en el manejo actual del control y registro de la información, para poder identificar de que otros flujos depende el resultado de listas de historia basada en problemas esto beneficio a tener una mejor idea de todos los datos que se debía considerar al momento de definir las funcionalidades del sistema.

4.4.2.4. Resultados de la etapa de análisis.

Luego de analizar la información que interviene en el manejo de Formularios se definieron los requerimientos del sistema, los mismos que fueron aprobados y se detallan a continuación:

-Requisitos Futuros

Debido a que el sistema está en etapa de desarrollo, el análisis de requerimientos futuros podrá realizarse una vez que el sistema este puesto en marcha, y si los directivos desean agregar mejoras o incluir nuevos módulos en el sistema se deberá realizar el estudio oportuno por parte de las personas que sean designadas dentro de la siguiente fase de e-Salud para cumplir con dicha necesidad.

-Requisitos Específicos

En este apartado se presentan los requisitos funcionales que deberán ser satisfechos por el sistema. Todos los requisitos aquí expuestos son esenciales, es decir, no sería aceptable un sistema que no satisfaga alguno de los requisitos aquí presentados. Estos requisitos se han especificado teniendo en cuenta, entre otros, el criterio de testeabilidad: dado un requisito, debería ser fácilmente demostrable si es satisfecho o no por el sistema.

-Requisitos funcionales

Para los requisitos siguientes el sistema permitirá al médico:

- Ingresar al sistema: Ingresar al sistema, luego de digitar su nombre de usuario, una contraseña y seleccionar el rol de acceso. Si el médico digitó correctamente dichos datos, podrá navegar entre las diferentes opciones.

- Registrar Ficha Familiar: Visualizará una pantalla donde podrá consultar por número de cédula y si el paciente ya se encuentra en el sistema se le cargara la información. Caso contrario tendrá que digitar todos los campos obligatorios que el sistema le pide para poder ser ingresado en la base de datos del MSP.
- Registrar Mortalidad Familiar: Ingresará todos los datos por pantalla algunos campos serán obligatorios por lo que si el medico no ingresa ese campo presentara un mensaje de error y no lo dejara ingresar la información a la base de datos del MSP.
- Registrar Actividad de Diagnóstico Familiar: Ingresará todos los datos por pantalla algunos campos serán obligatorios por lo que si el medico no ingresa ese campo presentara un mensaje de error y no lo dejara ingresar la información a la base de datos del MSP.
- Registrar Evaluación de las Intervenciones: Ingresará todos los datos por pantalla algunos campos serán obligatorios por lo que si el medico no ingresa ese campo presentara un mensaje de error y no lo dejara ingresar la información a la base de datos del MSP.

Requisitos no funcionales:

Software base para el desarrollo: El software base para el desarrollo del Formulario anexo1 Ficha Familiar será el IDE Eclipse debido a su gran potencial y que no posee licencia para su utilización.

El lenguaje de programación sobre el que se va a desarrollar el sistema de información será java porque es rápido, seguro, fiable y por sus potentes funcionalidades para la construcción.

4.4.2.5. Grupo Focal.

Debido que la entrevista solo se la puede aplicar a un grupo específico de profesionales de la salud que están autorizados por el Dr. Ronald Alvarado líder de proyecto de programa Promeinfo, para evaluar, dar sus opiniones sobre la funcionalidad y diseño de la aplicación e-Salud se utilizará la técnica de grupo focal.

Las personas que participan dentro de este grupo son del equipo básico de salud, tienen como características ser profesionales de la salud que realizan visitas a la comunidad. Ellos ofrecen charlas, recopilan información de fichas familiares, realizan mingas, visitan personas que no pueden acudir a un establecimiento de salud por su condición física o mental, visitan escuelas, realizan capacitaciones a mujeres embarazadas, etc.

Por motivos de que el proyecto debe tener unos altos parámetros de confidencialidad el MSP ha designado diez profesionales de la salud que han sido autorizados para poder dar su opinión acerca de la aplicación e-Salud. Los Doctores son los siguientes:

- Dr. Eloy Rivera
- Dr. César Navas
- Dra. Odont. Cecilia Molina
- Dra. Geomayra Delgado
- Dr. Luis Bueno
- Dr. Patricio Masaquiza
- Lcda. Mónica Guzmán
- Obst. Mayer Plúas
- Dra. Mónica Franco
- Dr. William Medina

Después de la reunión en la que se presenta el desarrollo del proyecto, las explicaciones globales y específicas del sistema, la interacción del grupo con el sistema, la revisión de las observaciones realizadas en el levantamiento de la información presentes en el trabajo terminado, concluimos con un pequeño test de aceptación del sistema mediante el uso del escalamiento Likert.

Preguntas de Aceptación del Proyecto

1.- ¿Está de acuerdo en que se implemente una herramienta tecnológica para las fichas familiares?

- a) Estoy de acuerdo
- b) Ni de acuerdo ni en desacuerdo
- c) Estoy en desacuerdo

2.- ¿Cómo califica su conocimiento y dominio de la computadora?

- a) Excelente
- b) Regular
- c) Malo

3.- ¿La aplicación e-Salud le permite interactuar de forma rápida?

- a) Siempre
- b) Algunas Veces
- c) Nunca

4.- ¿El ingreso de información por medio de la aplicación le resulta fácil?

- a) Siempre
- b) Algunas Veces
- c) Nunca

5.- ¿Qué le parece el diseño de las pantallas de ficha familiar?

- a) Excelente
- b) Regular
- c) Malo

- 6.- ¿Las pantallas de fichas familiares cumplen con los estándares del MSP?
- a) Estoy de acuerdo
 - b) Ni de acuerdo ni en desacuerdo
 - c) Estoy en desacuerdo
- 7.- ¿Cree usted que la aplicación ayuda optimizar el tiempo de atención del paciente?
- a) Estoy de acuerdo
 - b) Ni de acuerdo ni en desacuerdo
 - c) Estoy en desacuerdo
- 8.- ¿Es segura la interoperabilidad de los establecimientos de salud?
- a) Estoy de acuerdo
 - b) Ni de acuerdo ni en desacuerdo
 - c) Estoy en desacuerdo
- 9.- ¿Cómo considera que el ingreso de información sirva para los censos del país?
- a) Excelente
 - b) Regular
 - c) Mala
- 10.- ¿Cómo evalúa la calidad de la aplicación e-Salud?
- a) Excelente
 - b) Regular
 - c) Mala

Escalamiento Likert

Se hace sobre la base de afirmaciones tajantes, sobre las que el entrevistado da su opinión dentro de una gama de alternativas que van del total acuerdo al total desacuerdo. Para nuestro caso realizamos tres opciones de respuesta donde se va considerar las puntuaciones de la siguiente manera:

- a) = 3
- b) = 2
- c) = 1

Tendremos que nuestro puntaje máximo será 30 y nuestra nota mínima será 10.

Comenzaremos con los puntajes de los entrevistados a continuación:

-Entrevistado 1

De acuerdo a los resultados obtenidos del primer entrevistado la calificación fue la siguiente:

$$(3+2+3+3+2+2+3+2+2+3) = 25 \text{ puntos}$$

Se puede determinar que el primer entrevistado ve favorable y acepta de trabajar con una herramienta tecnológica.

-Entrevistado 2

De acuerdo a los resultados obtenidos del segundo entrevistado la calificación fue la siguiente:

$$(3+1+3+2+2+2+3+2+2+2) = 22 \text{ puntos}$$

Se puede determinar que el segundo entrevistado cree que la implementación de una herramienta tecnológica ayudará en los tiempos de aceptación, pero que le costara debido que no tiene mucho conocimiento con el computador.

Entrevistado 3

De acuerdo a los resultados obtenidos del tercer entrevistado la calificación fue la siguiente:

$$(3+2+3+2+3+2+3+3+2+3) = 26 \text{ puntos}$$

Se puede determinar que el tercer entrevistado apuesta por un cambio hacia la tecnología que beneficie a los pacientes de la comunidad.

Entrevistado 4

De acuerdo a los resultados obtenidos del cuarto entrevistado la calificación fue la siguiente:

$$(3+1+3+2+3+2+3+3+2+3) = 25 \text{ puntos}$$

Se puede determinar que el cuarto entrevistado le agrado la aplicación e-Salud pero quiere ser capacitado por no tener mucha experiencia con el computador.

Entrevistado 5

De acuerdo a los resultados obtenidos del quinto entrevistado la calificación fue la siguiente:

$$(2+2+2+2+2+2+2+3+2+3) = 22 \text{ puntos}$$

Se puede determinar que el quinto entrevistado está en un punto que le es indiferente cambiar del proceso manual al proceso automático que quiere realizar el MSP.

Entrevistado 6

De acuerdo a los resultados obtenidos del sexto entrevistado la calificación fue la siguiente:

$$(3+3+3+3+3+3+3+3+3+3) = 30 \text{ puntos}$$

Se puede determinar que el sexto entrevistado le gusto el sistema e-Salud y cree que existe un gran beneficio para los pacientes y para ellos en los cupos diarios que tienen que llegar de atención.

Entrevistado 7

De acuerdo a los resultados obtenidos del séptimo entrevistado la calificación fue la siguiente:

$$(3+1+2+3+2+2+2+2+3+3) = 23 \text{ puntos}$$

Se puede determinar que el séptimo entrevistado visualiza que la aplicación los ayudara en su día a día de trabajo con la comunidad pero dio unas observaciones sobre el diseño de las pantallas.

Entrevistado 8

De acuerdo a los resultados obtenidos del octavo entrevistado la calificación fue la siguiente:

$$(3+1+2+3+2+3+2+3+3+3) = 25 \text{ puntos}$$

Se puede determinar que el octavo entrevistado da su confianza que el MSP entregue una nueva herramienta de trabajo a los profesionales de la salud para brindar un mejor servicio a la comunidad.

Entrevistado 9

De acuerdo a los resultados obtenidos del noveno entrevistado la calificación fue la siguiente:

$$(3+2+3+2+3+3+2+3+2+3) = 26 \text{ puntos}$$

Se puede determinar que el noveno entrevistado está de acuerdo con que la aplicación visualizada servirá de mucha ayuda en los trabajos que tienen que realizar con la comunidad.

Entrevistado 10

De acuerdo a los resultados obtenidos del décimo entrevistado la calificación fue la siguiente:

$$(3+3+2+3+2+2+2+3+3+3) = 26 \text{ puntos}$$

Se puede determinar que el décimo entrevistado le llamo mucho la atención de poder realizar intercambio de información entre diferentes establecimientos de salud, lo cual nos comentó serviría en los casos de emergencias, consultas externas, y en no tener que duplicar información del paciente entre diferentes establecimientos de salud.

4.5. Entregables del proyecto

Los entregables de este proyecto de acuerdo a la metodología Scrum son:

- Código Fuente
- Código Ejecutable
- Manual de Usuario
- Manual Técnico

5. Capítulo IV

5.1. Criterios de aceptación del producto o servicio

Para poder justificar y aprobar el desarrollo del proyecto se determina realizar las pruebas internas de funcionalidad, en cuanto al acceso y el ingreso de información del sistema sobre el formulario de fichas familiares.

Cuadro N° 9 – Pruebas de funcionalidad

PRUEBAS	EFFECTIVIDAD
Validación de ingreso del Usuario al Sistema e-Salud 2016.	100%
Validación de Usuario por roles de acceso al Formulario Ficha Familiar.	100%
Validación de ingreso de cédula de la persona en el Formulario Ficha Familiar.	100%
Consulta si la cédula está en el Sistema para que se visualice el formulario de Ficha Familiar con los datos de la persona.	100%
Validación de los campos del formulario Ficha Familiar que se deben llenar de forma obligatoria con información de la persona.	100%
Validación del tipo de dato (Carácter, numérico, fecha) que se ingresa del formulario Ficha Familiar con información de la persona.	100%
Ingresa la información del cliente en el formulario de la Mortalidad de la Ficha Familiar.	100%
Ingresa la información del cliente en el formulario de Actividad de Diagnóstico de la Ficha Familiar.	100%
Ingresa la información del cliente en el formulario de Evaluación de las Intervenciones de la Ficha Familiar.	100%

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de investigación

Cuadro N° 10 – Matriz de criterios de aceptación

TEMA	DESARROLLO E IMPLEMENTACIÓN DEL FORMULARIO ANEXO 1 - FICHA FAMILIAR DEL MINISTERIO DE SALUD PÚBLICA, APLICANDO ARQUETIPOS BASADOS EN LA NORMA ISO-13606 PARA OBTENER INTEROPERABILIDAD ENTRE LOS SISTEMAS HOSPITALARIOS.		
OBJETIVO GENERAL	Desarrollar el formulario Ficha Familiar dispuesto por el Ministerio de Salud Pública para el Sistema de Información de Historia Clínica con la integración de arquetipos basados en la norma ISO 13606 para obtener interoperabilidad entre los sistemas hospitalarios, con el propósito de mejorar la calidad en atención a los usuarios.		
OBJETIVO ESPECÍFICO	OBJETIVO 1	OBJETIVO 2	OBJETIVO 3
	Desarrollar una aplicación para generación del formulario de Ficha Familiar propuesto por el MSP como parte de la Historia Clínica Electrónica que será implementado mediante lenguajes de programación y herramientas de última generación.	Diseñar y levantar línea base para la creación de los arquetipos con profesionales médicos en función al Formulario de Ficha Familiar, haciendo uso de la norma ISO 13606, lo que permitirá interoperabilidad entre los diferentes sistemas médicos existentes.	Realizar las pruebas de software a nivel local detalladas en un plan de pruebas para verificar y garantizar que el sistema de información funcione correctamente.
CRITERIO DE ACEPTACIÓN	Entrevista realizada a profesionales médicos para una mejor comprensión del requerimiento.	Entrevista realizada a profesionales médicos para una mejor comprensión del requerimiento.	Entrevista realizada a profesionales médicos para una mejor comprensión del requerimiento.
TAREA	Enumerar los requerimientos necesarios para el formulario de Ficha Familiar. Levantar información de cada requerimiento de los formularios del MSP.	Crear los métodos necesarios para aplicar interoperabilidad del formulario de Ficha Familiar que debe existir entre los diferentes establecimientos de salud pública y privada.	Realizar la respectiva investigación para escoger la mejor arquitectura que se debe aplicar en este Proyecto considerando la Norma ISO 13606 para beneficio de la comunidad.
MÉTODO	Investigar por los medios de internet, libros, papers, entrevistas y reuniones con el Dr. Eloy Rivera, Dr. Ronald Alvarado y el Ing. Jorge Medina.	Investigar por los medios de internet, entrevistas y reuniones con los Ing. Jorge Medina e Ing. Bernardo Iñiguez.	Realizar un documento de un plan de pruebas internas de toda la funcionalidad del formulario de Ficha Familiar.
TIEMPO	5 semanas	3 semanas	1 semanas
RECURSOS	Intervienen en el Proyectos dos desarrolladores de Sistemas Carlos Medina – Fernando Flores.	Intervendrán en el Proyectos dos desarrolladores de Sistemas Carlos Medina – Fernando Flores.	Intervendrán en el Proyectos dos desarrolladores de Sistemas Carlos Medina – Fernando Flores.
CONCLUSIONES	El tener un sistema interoperable que pueda ser ingresado desde cualquier establecimiento público o privado es una ventaja para los profesionales de salud y para la sociedad Ecuatoriana.	Se desarrolla el Sistema aplicando las Normas ISO 13066 para que los datos que se ingresen en el formulario tengan la validación de los arquetipos.	Se establece que los diferentes escenarios del Sistema con el formulario de Ficha Familiar han sido validados con el Usuario que maneja la herramienta tecnológica.
RECOMENDACIONES	Se debe implementar en un Servidor para que sea utilizado por parte del Ministerio de Salud Pública, así como también dar los respectivos roles de Usuario para que puedan ingresar a la opción correcta.	Realizar el respectivo ingreso de información de las fichas familiares para que los arquetipos puedan funcionar de forma óptima y poder compartir datos entre los establecimientos públicos y privados.	Revisar en Producción si existen escenarios nuevos que no fueron tomados en cuenta en el levantamiento de información de los Usuarios.

Elaborado por: Carlos Medina, Fernando Flores

Fuente: Datos de investigación

5.2. Conclusiones

Se pueden dar las siguientes conclusiones en el formulario de ficha familiar:

- El Sistema de ingreso de información del formulario de ficha familiar, logra la interoperabilidad entre distintos establecimientos de salud del país con el aval del MSP y además brinda los estándares de las normas ISO para dar un realce a la herramienta tecnológica y a la comunidad de Salud del Ecuador.
- Los establecimientos de salud tendrán una herramienta tecnológica que es robusta, fácil de manejar, además que se pueda usar desde cualquier lugar y dispositivo que tenga acceso al internet.
- Se pudo notar que el MSP debe realizar capacitaciones a los profesionales de Salud debido que no tienen mucho dominio con la computadora lo que hace que se les dificulte la interacción con la aplicación e-Salud.
- La aplicación e-Salud en esta primera fase de desarrollo está elaborada para que pueda trabajar sobre una aplicación web, en caso de ser necesario una aplicación en dispositivos móviles se debe ajustar en la programación.
- Se encontró que existen parámetros de estándares médicos que deben ser validados por los arquetipos que se van a utilizar en el sistema e-Salud.
- Se realizó de acuerdo con el plan de trabajo dispuesta en crear un mecanismo automático de validación de estándares médicos a través de los arquetipos impulsado por los médicos involucrados en el proyecto para poder cumplir con el objetivo del intercambio de información entre diferentes establecimientos de salud.

- Se implementó en conjunto con el equipo de desarrolladores, arquitectos de software, base de datos la integración del sistema e-Salud para poder realizar una Historia Clínica Única.
- Durante la Fase de implementación del proyecto e-Salud se realizó la etapa de pruebas internas que consiste en realizar las pruebas de estrés de las distintas funcionalidades de las pantallas que se automatizaron en la ficha familiar, en conjunto con el área de calidad.

5.3. Recomendaciones

Se pueden dar las siguientes recomendaciones en el formulario de ficha familiar:

- Se debe ingresar información real, debido a que los datos se compartirán entre diferentes establecimientos de salud.
- Se debe realizar para una segunda fase del proyecto la parte de la firma electrónica del supervisor responsable del ingreso de la información de la ficha familiar en la pantalla ingreso de datos de la vivienda.
- Para una nueva fase del proyecto se debe implementar el sistema en un servidor de producción del MSP, con el desarrollo y los estándares que se aplicó en esta primera fase para poder gozar de la funcionalidad para la sociedad Ecuatoriana.

- Se debe realizar un Módulo de Reportes para que los Usuarios puedan hacer la diferente toma de decisiones y cuadros estadísticos.

- El Ministerio de Salud Pública debe realizar las diferentes capacitaciones al personal que estará encargado del Sistema.

- En caso de salir nuevas definiciones estas deben ser tomadas en cuenta en una nueva fase de desarrollo y puesta en producción.

- Se debe tener en cuenta que existen campos obligatorios que deben ser ingresados en el formulario, en caso de no hacerlo no se puede continuar con la interacción en la pantalla de la ficha familiar.

- La aplicación e-Salud está diseñada para funcionar en un computador de escritorio o laptop, en caso de necesitar que el software trabaje sobre un dispositivo móvil (Tablet, celular, palm, etc) se requiere realizar cambios en la programación para que se pueda visualizar y trabajar de forma correcta.

Se establece escoger el código de enfermedad CIE en caso de ser necesario.

6. Bibliografía

- ADOBE. (2016). *Aspectos básicos de las aplicaciones*. Obtenido de <https://helpx.adobe.com/es/dreamweaver/using/web-applications.html>
- Chang Campos, C. (Agosto de 2007). *Consejo Nacional de Salud*. Obtenido de CONASA: http://www.conasa.gob.ec/codigo/publicaciones/hcu/historia_clinica.pdf
- Chang Campos, C. (14 de marzo de 2008). *Revista Judicial* . Obtenido de <http://www.derechoecuador.com/productos/producto/catalogo/registros-oficiales/2008/abril/code/19031/registro-oficial-no-316---martes-15-de-abril-de-2008#No0138>
- Congreso Nacional. (27 de Marzo de 1998). *Reg. Oficial 320 Ley de Propiedad Intelectual*. Obtenido de http://www.correosdeecuador.gob.ec/wp-content/uploads/downloads/2015/05/LEY_DE_PROPIEDAD_INTELECTUAL.pdf
- Congreso Nacional. (22 de Diciembre de 2006). *Ley 77 De Derechos y Amparo del Paciente*. Obtenido de <http://www.salud.gob.ec/wp-content/uploads/downloads/2014/09/Normativa-Ley-de-Derechos-y-Amparo-del-Paciente.pdf>
- Congreso Nacional. (24 de Enero de 2012). *Ley Orgánica de Salud*. Obtenido de http://www.desarrollosocial.gob.ec/wp-content/uploads/downloads/2015/04/SALUD-LEY_ORGANICA_DE_SALUD.pdf
- csi2altair. (2016). *Wikispaces*. Obtenido de <https://csi2altair.wikispaces.com/WildFly>
- De Pablos Heredero, C., López Hermoso Agius, J. J., Romo Romero, S. M., & Medina Salgado, S. (2011). *Organización y transformación de los sistemas de información en la empresa*. España: ESIC EDITORIAL.
- Durán, F., Gutiérrez, F., & Pimente, E. (2007). *Programación orientada a objetos con Java*. España: Thomson.
- Educación IT. (2013). *¿Qué es Java Hibernate?* Obtenido de <http://blog.educacionit.com/2013/02/07/que-es-java-hibernate/>
- Fuente, D. D. (14 de Julio de 2008). *Intramed*. Obtenido de <http://www.intramed.net/contenidover.asp?contenidoID=53769>

Fundación OpenEHR. (s. f.). *OpenEHR*. Obtenido de <http://www.openehr.org/about/foundation>

Gallardo, D. (2012). *Iniciándose en la plataforma Eclipse*. Obtenido de <http://www.ibm.com/developerworks/ssa/library/os-ecov/>

Guzmán, F., & Arias, C. A. (2012). La historia clínica: Elemento fundamental del acto médico. *SciELO*, 15-24. Obtenido de La historia clínica: Elemento fundamental del acto médico: <http://www.scielo.org.co/pdf/rcci/v27n1/v27n1a2.pdf>

HERRERA RAMIREZ, F. J. (2008). *MANUAL DE RESPONSABILIDAD MEDICA*. BOGOTÁ: LEYER.

INEN. (2008). *Instituto Ecuatoriano de Normalización*. Obtenido de http://www.normalizacion.gob.ec/wp-content/uploads/downloads/2014/NORMAS_2014/DRO/nte_inen_iso_13606_1extracto.pdf

Latorre, G. (2010). Obtenido de <http://gl-epn-programacion-ii.blogspot.com/2010/03/jym-jdk-jre-conceptos-fundamentales-de.html>

Letelier, & Penadés. (2006).

LinkEHR. (2005-2013). *LinkEHR Editor/Studio*, v20131012. (U. P. València, Productor) Obtenido de http://pangea.upv.es/new_linkehr_web/data/LinkEHRStudioManual.pdf

López, E. (s.f.). *Academia*. Obtenido de http://www.academia.edu/10102692/Arquitectura_de_n_capas

MARTÍNEZ VILLALOBOS, G., CAMACHO SÁNCHEZ, G., & BIANCHA GUTIÉRREZ, D. (2012). DISEÑO DE FRAMEWORK WEB PARA EL DESARROLLO DINÁMICO DE APLICACIONES. *Scientia Et Technica*, 178-183. Obtenido de <http://www.redalyc.org/articulo.oa?id=84917316032>

Martínez, R. (2009-2013). *PostgreSQL-es*. Obtenido de http://www.postgresql.org.es/sobre_postgresql

Ministerio de Salud Pública. (14 de Marzo de 2008). Reg. Oficial HCL Única. Quito, Ecuador.

Ministerio de Salud Pública. (s. f.). *Ministerio de Salud Pública de Ecuador*. Obtenido de <http://www.salud.gob.ec/valores-mision-vision/>

Ministerio de Salud Pública. (s. f.). Misión. Quito, Ecuador.

Ministra de Salud Pública. (12 de Abril de 2008). Reg. Oficial 316 Acuerdo 0138. Obtenido de <http://www.derechoecuador.com/productos/producto/catalogo/registros->

oficiales/2008/abril/code/19031/registro-oficial-no-316---martes-15-de-abril-de-2008#No0138

Ministro de Salud Pública. (29 de Enero de 2015). *Acuerdo Ministerial 5216 Reglamento de Información Confidencial en Sistema Nacional de Salud*. Obtenido de <http://instituciones.msp.gob.ec/cz6/images/lotaip/Enero2015/Acuerdo%20Ministerial%205216.pdf>

Moquillaza Henríquez, S. D., Guerra Grados, L., & Vega Huerta, H. (2010). Programación en N capas. *Revista de investigación de sistemas e informática*, 57-67. Obtenido de <http://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/3283/2741>

MSP. (12 de Octubre de 2012). *Ministerio de Salud Pública*. Obtenido de <http://instituciones.msp.gob.ec/somossalud/index.php/noticias-de-la-reforma/126-que-es-el-modelo-de-atencion-integral-de-salud>

Muñoz Carrero, A. (s.f.). *MSSSI*. Obtenido de http://www.msssi.gob.es/organizacion/sns/planCalidadSNS/docs/MUNOZ_CARRERO.pdf

Olivas, C. (2016). Interfaz de Persistencia Java (JPA) - Entidades y Managers. <http://www.lab.inf.uc3m.es/~a0080802/RAI/jpa.html>.

Oracle Corporation - Java. (2016). *Java*. Obtenido de https://www.java.com/es/download/faq/whatis_java.xml

Pardo, D. (2016). Programacion J2ee. <http://www.programacionj2ee.com/que-es-spring-framework/>.

Penadés, L. &. (2006).

Presidente Correa Delgado, R. (12 de Octubre de 2010). *Reg. Oficial 298 Ley Orgánica de Educación Superior*. Obtenido de http://planipolis.iiep.unesco.org/upload/Ecuador/Ecuador_Ley_organica_educacion_superior.pdf

Presidente Noboa Bejarano, G. (28 de Enero de 2003). *Decreto 3611 Reglamento Orgánico a la Ley Orgánica del Sistema Nacional de Salud*. Obtenido de <http://www.desarrollosocial.gob.ec/wp-content/uploads/downloads/2013/10/reglamento-sis-nac-salud.pdf>

Registro Oficial Suplemento 427. (29 de Enero de 2015). *Ministerio de Salud Pública*. Obtenido de

<http://instituciones.msp.gob.ec/cz6/images/lotaip/Enero2015/Acuerdo%20Ministerial%205216.pdf>

Rubio Gonzáles , M. D. (2009). Generación automática de aplicaciones para la Gestión de HCE. *digitum*, 14-43.

SIISE. (s. f.). *Sistema Integrado de Indicadores Sociales del Ecuador*. Obtenido de http://www.siise.gob.ec/siiseweb/PageWebs/Accion%20Social/ficacc_OS037.htm

Tallado San Miguel, J. (2015). *Implantación de aplicaciones web en entorno internet, intranet y extranet*. España: Nobel. Obtenido de <https://books.google.com.ec/books?id=RtESCgAAQBAJ&printsec=frontcover&hl=es#v=onepage&q&f=false>

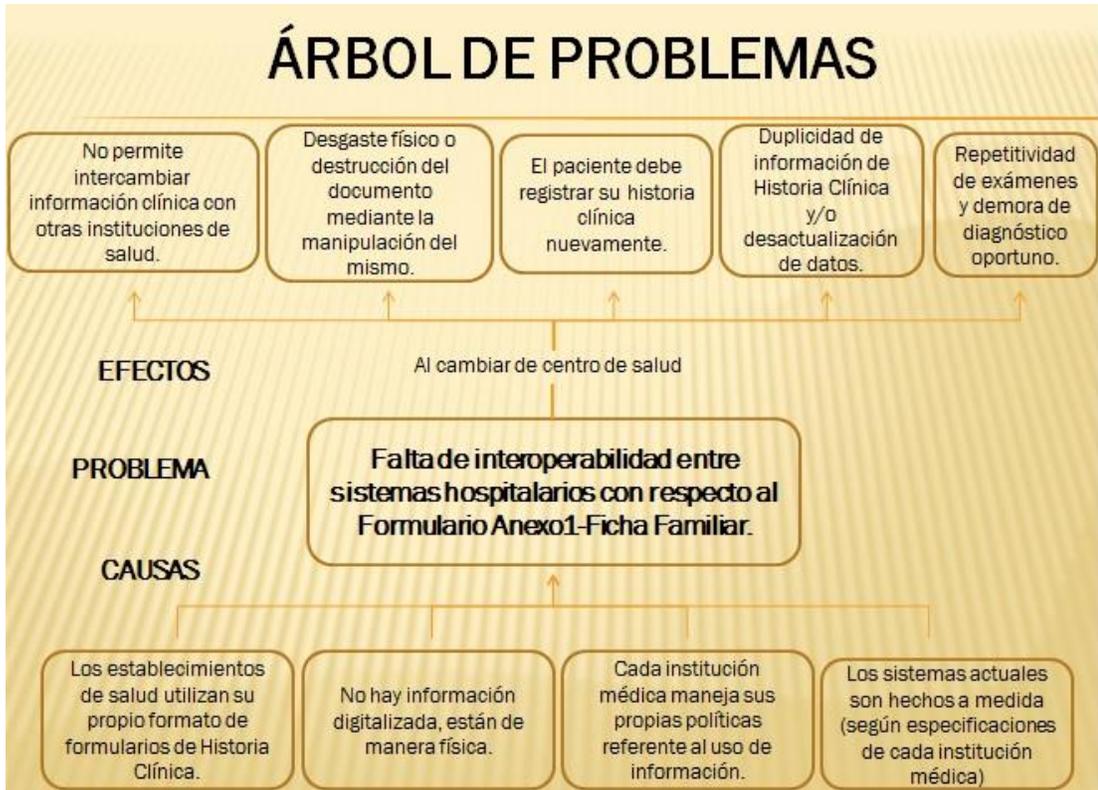
The Nemours Foundation. (2014). *KidsHealth*. Obtenido de <http://kidshealth.org/es/parents/ehrs-esp.html>

Universidad Politécnica de Valencia UPV. (2004-2015). *Plataforma LinkEHR*. Obtenido de <http://linkehr.com/>

Villalobos, G., Sánchez, G., & Gutiérrez, D. (2010). Diseño de framework web para el desarrollo dinámico de Aplicaciones. *Scientia Et Technica*, 178-183.

7. Anexos

7.1. Árbol de problemas



7.2. Árbol de objetivos



7.4. Formato ficha familiar hoja 3

PARA TODAS LAS PERSONAS		SECCIÓN V. DATOS DE LOS MIEMBROS DEL HOGAR				PARA MUJERES DE 12 A 49 AÑOS		
NACIONALIDAD		SEGUROS	SEGURO PRIVADO	PATERNICIDAD	EMBARAZO	NACIDOS VIVOS		
¿Qué nacionalidad tiene (1)?		() Esta Afiliación oculto por: ESS, Seguro General?.....1 ESS, Seguro Voluntario?.....2 ESS, Seguro Compasivo?.....3 Seguro ISSFA?.....4 Seguro ISSPOL?.....5 Ninguno?.....0	¿() Tiene seguro de salud privado?	¿() Tiene hijo o hijas?	¿Está () actualmente embarazada?	¿Es () hijo o hija NACIDOS VIVOS ya nacido () durante toda su vida?	¿Es () VIVO al ÚLTIMO HIJO(A) NACIDO(A) (VIVA) de (1)?	
00	Per	15	16	17	18	19	20	21
01								
02								
03								
04								
05								
06								
07								
08								
09								
10								
11								
12								

7.6. Formato ficha familiar hoja 8

HORA:

FECHA	CUMPLIMIENTO Y RESULTADOS			COMPROMISOS
	Código cumple	Código resultado	Causas y dificultades	
01 Día <input type="text"/> Mes <input type="text"/> Año <input type="text"/>				Detalle de los acuerdos alcanzados
02 Día <input type="text"/> Mes <input type="text"/> Año <input type="text"/>				
03 Día <input type="text"/> Mes <input type="text"/> Año <input type="text"/>				
04 Día <input type="text"/> Mes <input type="text"/> Año <input type="text"/>				
05 Día <input type="text"/> Mes <input type="text"/> Año <input type="text"/>				

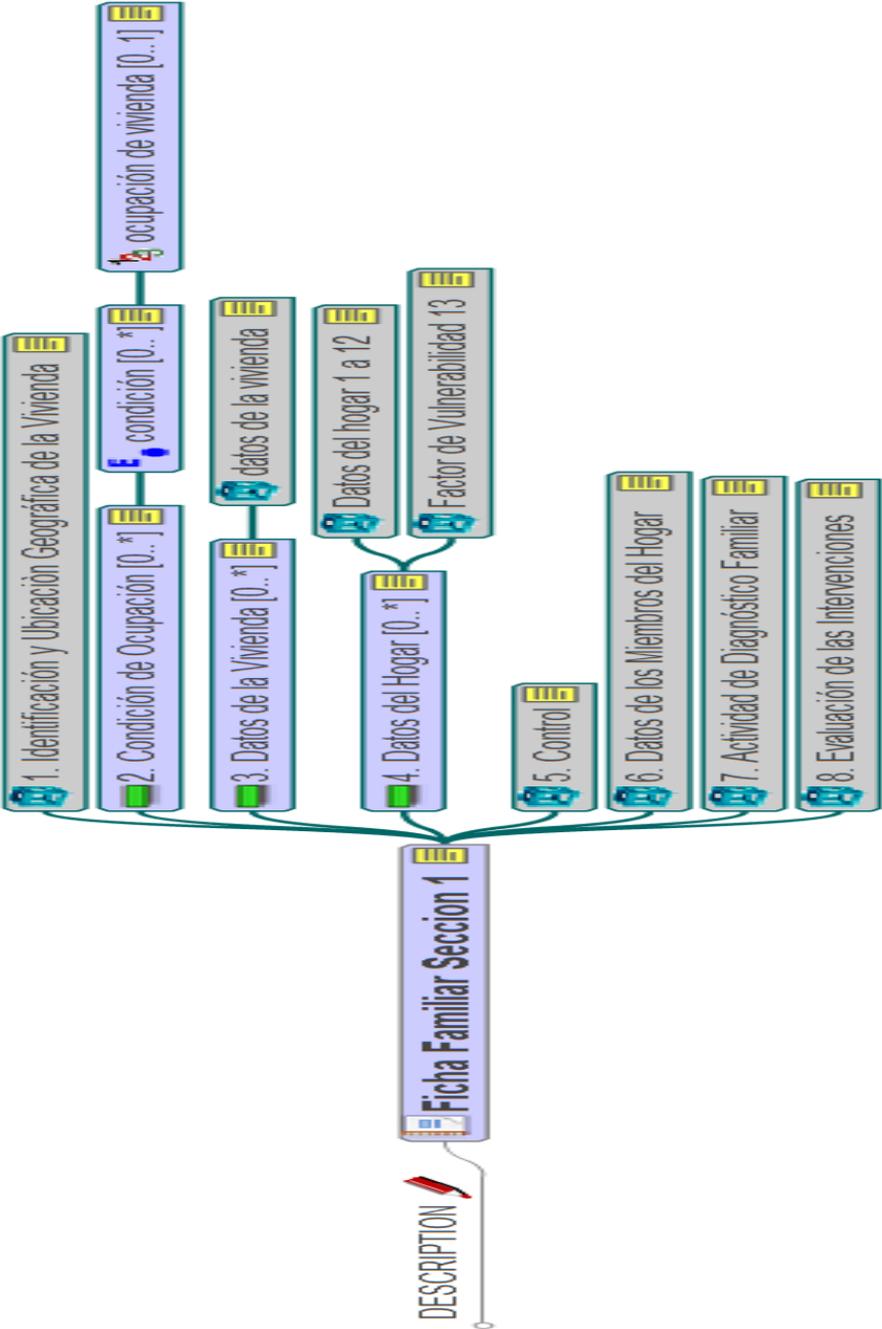
7.7. Ejemplo hoja1 sección 1 creando arquetipo

The screenshot displays the CEN-EN13606 Editor interface. The main window is titled 'CEN-EN13606-SECTION Identificación y Ubicación Geográfica de la Vivienda.v1' and shows the 'ARCHETYPE' editor. The interface is divided into several sections:

- Header:** Identifies the archetype as 'CEN-EN13606-SECTION Identificación y Ubicación Geográfica de la Vivienda.v1'.
- Specializes:** A list of related archetypes, including 'Dirección', 'Identificación Censal', 'Identificación de Sistema y Ubicación', 'Cambio', 'Distrito', 'Instalación del Sistema', 'Parroquia', 'Provincia', and 'Tipo Transporte y Tiempo Desplazamiento'.
- Concepts:** 'Identificación y Ubicación Geográfica de la Vivienda'.
- Lifecycle State:** 'Draft'.
- Version:** '1'.
- Original Language:** 'es-ec'.
- Details (es-ec):** Fields for 'Copyright' and 'Keywords'.
- Purpose:** A text area for describing the archetype's purpose.
- Table:** A table with columns for 'Name' and 'Code'. It contains one row with the header 'Name' and 'Code'.

The bottom of the window shows a 'Node Info' panel with 'Terminology' selected, and a 'Copy Name' and 'Copy Code' button.

7.8. Ejemplo Diagrama de Arquetipo General



7.9. Ejemplo hoja1 sección I árbol de arquetipo

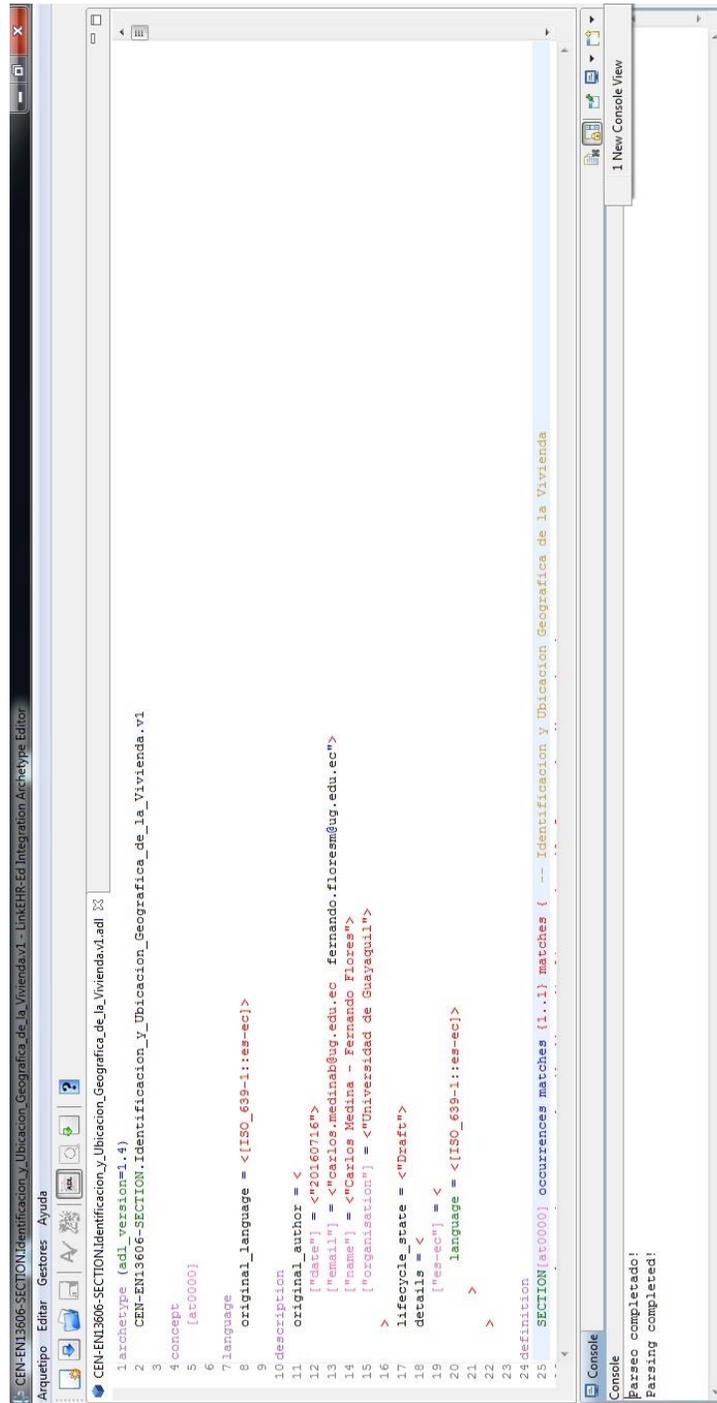
The screenshot displays the Archetype Editor interface. On the left, the 'Arquetype Tree' shows a hierarchical structure of concepts and their relationships. The main window is titled 'ARCHETYPE' and contains the following information:

- Header:**
 - Identifier: CEN-EN13606-SECTION.Identificacion_y_Ubicacion_Geografica_de_la_Vivienda.v1
 - Specializes: Identification y Ubicacion Geografica de la Vivienda
 - Concept: Draft
 - Lifecycle State: 1
 - Version: 1
 - Original Language: es-ec
- Details (es-ec):**
 - Copyright:
 - Keywords:
 - Purpose:
- Navigation:** Node Info, Console, Terminology
- Table:** A table with columns for Name, Code, Copy Name, and Copy Code.

The 'Arquetype Tree' on the left shows the following structure:

- CEN-EN13606-SECTION.Identificacion_y_Ubicacion_Geografica_de_la_Vivienda.v1 (es-ec)
 - Language
 - Ontology
 - Definition
 - members [0..5] (Obligatory)
 - Identificacion y Ubicacion Geografica de la Vivienda (1)
 - members [0..5] (Obligatory)
 - Direccion (0..1)
 - items [0..10] (Obligatory)
 - Calle Principal (0..1)
 - Calle Transversal (0..1)
 - Lote N° (0..1)
 - Departamento N° (0..1)
 - Nombre o Numero del Conjunto Habitacional (0..1)
 - Numero Celular (0..1)
 - Numero Convencional (0..1)
 - N° (0..1)
 - Piso N° (0..1)
 - Referencia Ubicacion de Vivienda (0..1)
 - Identificacion Censal (0..1)
 - items [0..3] (Obligatory)
 - Localidad/Comunidad/Recinto/Barrio (0..1)
 - Sector (0..1)
 - Zona (0..1)
 - Identificacion de Sistema y Ubicacion (0..1)
 - items [0..5] (Obligatory)
 - Canton (0..1)
 - Distrito (0..1)
 - Institucion del Sistema (0..1)
 - Parroquia (0..1)
 - Provincia (0..1)
 - Tipo, Transporte y Tiempo Desplazamiento (0..1)

7.10. Ejemplo hoja1 sección1 vista archivo ADL



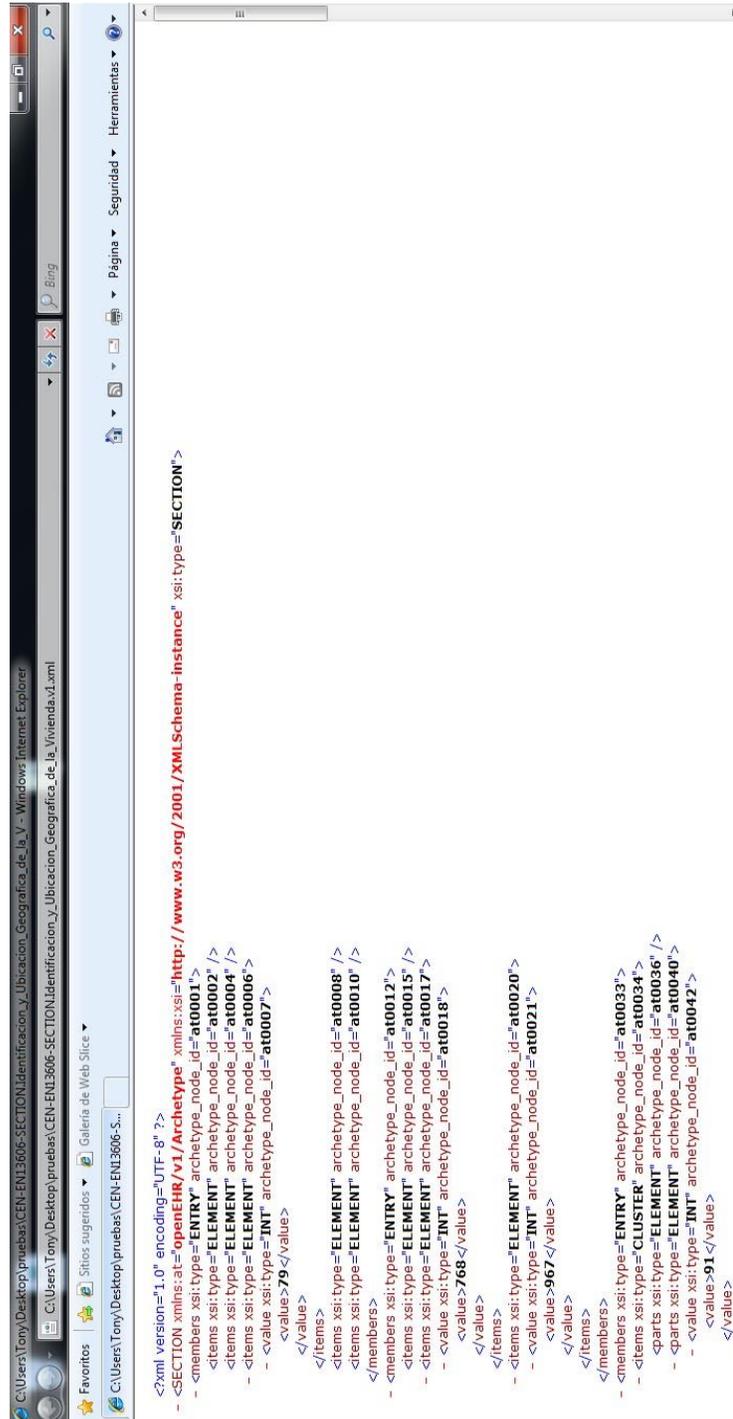
The screenshot shows an IDE window titled "CEN-EN13606-SECTION_Identificacion_Y_Ubicacion_Geografica_de_la_Vivienda.v1 - UnixEHR-Ed Integration Archetype Editor". The main editor area displays the following ADL content:

```
1 archetype (adl version=1.4)
2   CEN-EN13606-SECTION_Identificacion_Y_Ubicacion_Geografica_de_la_Vivienda.v1
3
4 concept
5   {at0000}
6
7 language
8   original_language = <[ISO_639-1::es-ec]>
9
10 description
11   original_author = <
12     ["date"] = <"20160716">
13     ["email"] = <"Carlos.medina@ug.edu.ec fernando.flores@ug.edu.ec">
14     ["name"] = <"Carlos Medina - Fernando Flores">
15     ["organisation"] = <"Universidad de Guayaquil">
16   >
17   lifecycle_state = <"Draft">
18   details = <
19     ["es-ec"] = <
20       language = <[ISO_639-1::es-ec]>
21     >
22   >
23
24 definition
25   SECTION[at0000] occurrences matches {1..1} matches { -- Identificacion Y Ubicacion Geografica de la Vivienda
```

The console window at the bottom shows the following output:

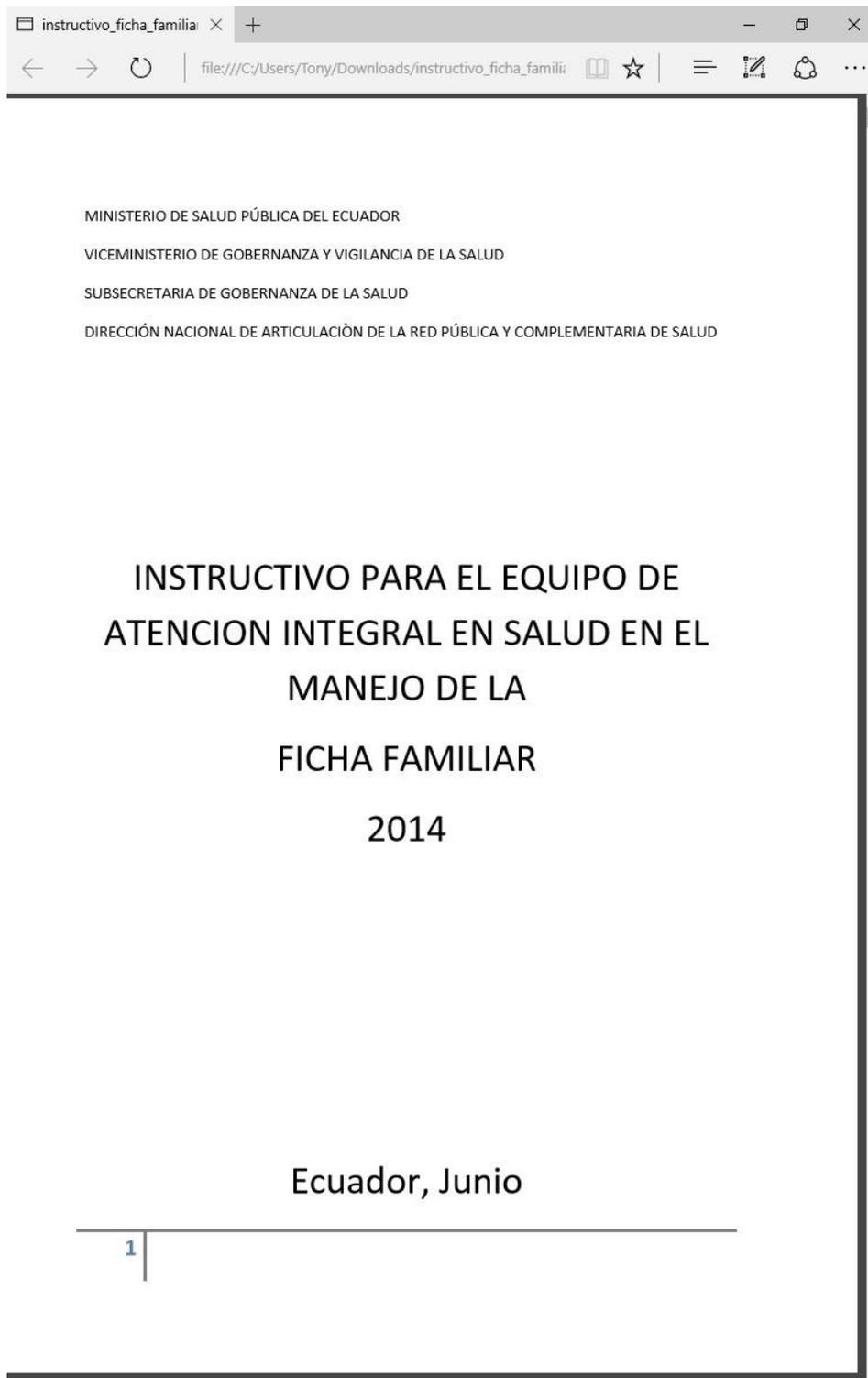
```
Console
Parseo completado!
Parsing completed!
```

7.11. Ejemplo hoja1 sección1 vista archivo XML



```
<?xml version="1.0" encoding="UTF-8" ?>
-<SECTION xmlns:at="openh3r/v1/Archetype" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="SECTION">
  <members xsi:type="ENTRY" archetype_node_id="at0001" />
  <items xsi:type="ELEMENT" archetype_node_id="at0002" />
  <items xsi:type="ELEMENT" archetype_node_id="at0004" />
  <value xsi:type="INT" archetype_node_id="at0006">
    <value>79</value>
  </value>
</items>
<items xsi:type="ELEMENT" archetype_node_id="at0008" />
<items xsi:type="ELEMENT" archetype_node_id="at0010" />
</members>
-<members xsi:type="ENTRY" archetype_node_id="at0012">
  <items xsi:type="ELEMENT" archetype_node_id="at0015" />
  <items xsi:type="ELEMENT" archetype_node_id="at0017">
    <value xsi:type="INT" archetype_node_id="at0018">
      <value>768</value>
    </value>
  </items>
  <items xsi:type="ELEMENT" archetype_node_id="at0020">
    <value xsi:type="INT" archetype_node_id="at0021">
      <value>967</value>
    </value>
  </items>
</members>
-<members xsi:type="ENTRY" archetype_node_id="at0033">
  <parts xsi:type="ELEMENT" archetype_node_id="at0034">
    <parts xsi:type="ELEMENT" archetype_node_id="at0036" />
    <value xsi:type="INT" archetype_node_id="at0040">
      <value>91</value>
    </value>
  </parts>
</members>
</SECTION>
```

7.12. Instructivo para manejo de Ficha Familiar



7.13. Entrevista preguntas de aceptación del proyecto

- 1.- ¿Está de acuerdo en que se implemente una herramienta tecnológica para las fichas familiares?
a) Estoy de acuerdo b) Ni de acuerdo ni desacuerdo c) Estoy en desacuerdo
- 2.- ¿Cómo califica su conocimiento y dominio de la computadora?
a) Excelente b) Regular c) Malo
- 3.- ¿La aplicación e-Salud le permite interactuar de forma rápida?
a) Siempre b) Algunas veces c) Nunca
- 4.- ¿El ingreso de información por medio de la aplicación le resulta fácil?
a) Siempre b) Algunas veces c) Nunca
- 5.- ¿Qué le parece el diseño de las pantallas de ficha familiar?
a) Excelente b) Regular c) Malo
- 6.- ¿Las pantallas de fichas familiares cumplen con los estándares del MSP?
a) Estoy de acuerdo b) Ni de acuerdo ni desacuerdo c) Estoy en desacuerdo
- 7.- ¿Cree usted que la aplicación ayuda optimizar el tiempo de atención del paciente?
a) Estoy de acuerdo b) Ni de acuerdo ni desacuerdo c) Estoy en desacuerdo
- 8.- ¿Es segura la interoperabilidad de los establecimientos de salud?
a) Estoy de acuerdo b) Ni de acuerdo ni desacuerdo c) Estoy en desacuerdo
- 9.- ¿Cómo considera que el ingreso de información sirva para los censos del país?
a) Excelente b) Regular c) Mala
- 10.- ¿Cómo evalúa la calidad de la aplicación e-Salud?
a) Excelente b) Regular c) Mala