



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD: INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS
CARRERA: SOFTWARE

GUÍA DE LABORATORIO DE APLICACIONES INFORMÁTICAS II

PRÁCTICA No.

1. DATOS GENERALES:

NOMBRE DEL ESTUDIANTE

David Brito

Blanca Proaño

CODIGO DEL ESTUDIANTE

6556

6464

FECHA DE REALIZACIÓN:

2023/04/22

FECHA DE ENTREGA:

2023/04/23

2. OBJETIVO(S):

2.1. GENERAL

- Construir la capa de presentación de la aplicación Polimed.

2.2. ESPECÍFICOS

- Diseñar y desarrollar una interfaz de usuario intuitiva y fácil de usar, para brindar una experiencia positiva al usuario.
- Implementar elementos interactivos, como formularios, botones y otros controles, para permitir que los usuarios interactúen eficientemente con la aplicación.
- Mantener una apariencia y comportamiento visual coherentes en toda la aplicación para que los usuarios puedan entender y prever cómo interactuar en diferentes partes de la interfaz.
- Mostrar datos de manera clara y comprensible utilizando gráficos, tablas u otros elementos visuales que faciliten la interpretación de la información.

- Realizar validaciones de datos en el lado del cliente para mejorar la experiencia del usuario y reducir la carga en el servidor al evitar la presentación de datos incorrectos o incompletos.

3. METODOLOGÍA

Tomando en cuenta los requisitos y necesidades del usuario en cuanto a lo que necesita, se realizaron diferentes mockups de las vistas de las interfaces, como una base para generar las vistas finales que llevará la aplicación.

En este caso para realizarlo se ah optado por el uso de tecnologías como Figma y Justmind, las cuales son dos herramientas que realmente facilitan el diseño, con sus diferentes herramientas.

Una vez finalizado, el diseño y validados por el usuario se ah procedido al diseño de las interfaces, pero tomando en cuenta un desarrollo incremental, es decir implementar y probar una por una las interfaces, para facilitar la identificación temprana de problemas y poder ajustar continuamente según sea necesario.

Al ser una aplicación con la arquitectura basada en capas, la capa de presentación se encuentra integrada con la capa lógica, en la cual se encuentra instaurada la lógica de negocio y también la comunicación con la capa de acceso a datos. A su vez el presente documento, realizado para documentar la codificación de la capa de presentación, los procesos y decisiones de diseño, para facilitar la comprensión y a futuro generar una versión completa eh instaurada en la institución requerida.

4. EQUIPOS Y MATERIALES:

- Computador
- Entorno integrado de desarrollo (IDE)
- Aula virtual
- Acceso a internet
- Repositorio
- Bibliografía

5. MARCO TEORICO:

La gestión eficiente de citas médicas es esencial para la operación suave de instituciones de salud. Este proceso involucra la programación, seguimiento y coordinación de citas entre pacientes y proveedores de atención médica. La optimización de este proceso no solo mejora la eficiencia operativa sino también la satisfacción del paciente.

Además, las interfaces de usuario en aplicaciones de salud desempeñan un papel crítico en la experiencia del usuario. La usabilidad y accesibilidad son aspectos clave para garantizar que los usuarios, incluidos pacientes y profesionales de la salud, puedan interactuar de manera efectiva con la aplicación. Un diseño centrado en el usuario se traduce en una mayor adopción y cumplimiento de los usuarios.

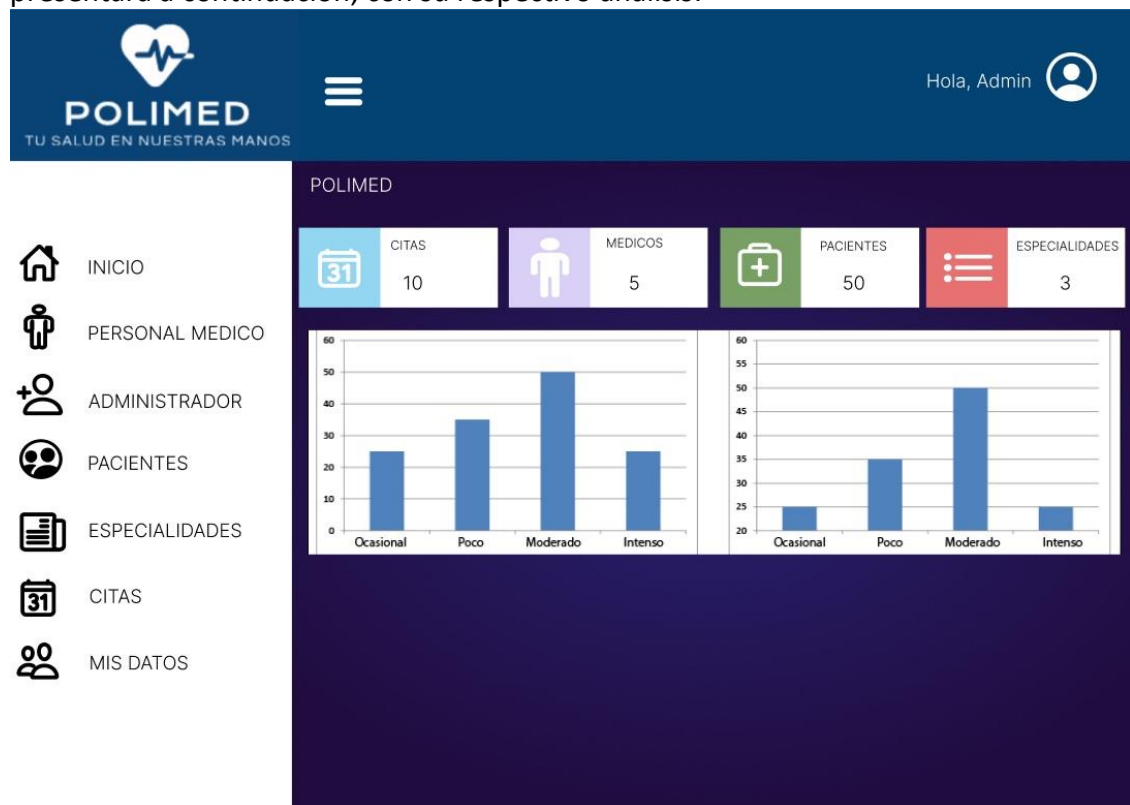
La usabilidad se refiere a la eficacia y eficiencia con que los usuarios pueden realizar tareas en la aplicación, mientras que la UX aborda la satisfacción general del usuario. Un diseño intuitivo, una navegación clara y una presentación de información comprensible contribuyen a una experiencia positiva del usuario en el contexto de la gestión de citas médicas.

Tomando en cuenta lo anterior y que se trata de una arquitectura basada en capas la interfaz de usuario se comunica con la capa de negocios para gestionar citas médicas. La selección cuidadosa de tecnologías y herramientas, como frameworks de desarrollo frontend, bibliotecas de UI y herramientas de diseño, impacta en la eficiencia del desarrollo y en la calidad de la interfaz de usuario resultante. En este caso para el desarrollo se utilizaron tecnologías como el IDE Netbeans, la unidad de persistencia EclipseLink, Figma, entre otras tecnologías.

Métodos como pruebas de usabilidad, encuestas y análisis de métricas de interacción permiten evaluar y mejorar continuamente la experiencia del usuario en la aplicación, las cuales serán tomadas en la próxima versión con enfoque más amplio.


6. PROCEDIMIENTO:

Al disponer de la creación tanto de la capa de enlace de datos como de la capa lógica de negocios, se procedió, tomando en cuenta la búsqueda del enfoque de versatilidad, usabilidad y rentabilidad, para generar mockups amigables para el usuario, como se presentará a continuación, con su respectivo análisis.




En el presente mockup, se puede observar la vista de administrador la cual se puede observar la vista del administrador, contando con un menú lateral que dispone de, un botón de inicio, el cual mostrará, como se puede observar el número de citas, la cantidad de médicos, la cantidad de pacientes y el número de especialidades.








Un botón de personal medico el cual desplegará la siguiente vista:



POLIMED
TU SALUD EN NUESTRAS MANOS


Administrador





 INICIO
  PERSONAL MEDICO
  PACIENTES
  CITAS
  ESPECIALIDADES
  ADMINISTRADOR
  MIS DATOS

PERSONAL MEDICO

NUEVO




BUSCAR


APELLIDOS	NOMBRES	ESPECIALIDAD	ACCION
sample text	sample text	sample text	 
sample text	sample text	sample text	sample text
sample text	sample text	sample text	sample text
sample text	sample text	sample text	sample text

En la cual se realizará la gestión respectiva del personal médico, es decir el registró, actualización, modificación de datos u eliminación de los mismos, a la vez que como se puede observar, el disponer de la información de la especialidad en la que está encargado cada uno.


El botón de pacientes, nos brindará la siguiente vista:










POLIMED
TU SALUD EN NUESTRAS MANOS



Administrador



 INICIO
  PERSONAL MEDICO
  ADMINISTRADOR
  PACIENTES
  ESPECIALIDADES
  CITAS
  MIS DATOS

Registro de Datos Paciente

Cédula de Ciudadanía

Nombres

Apellidos

Género

☐ Hombre
 ☐ Mujer

Fecha de Nacimiento

Edad 29

Dirección




Teléfono

E-mail

Enfermedad

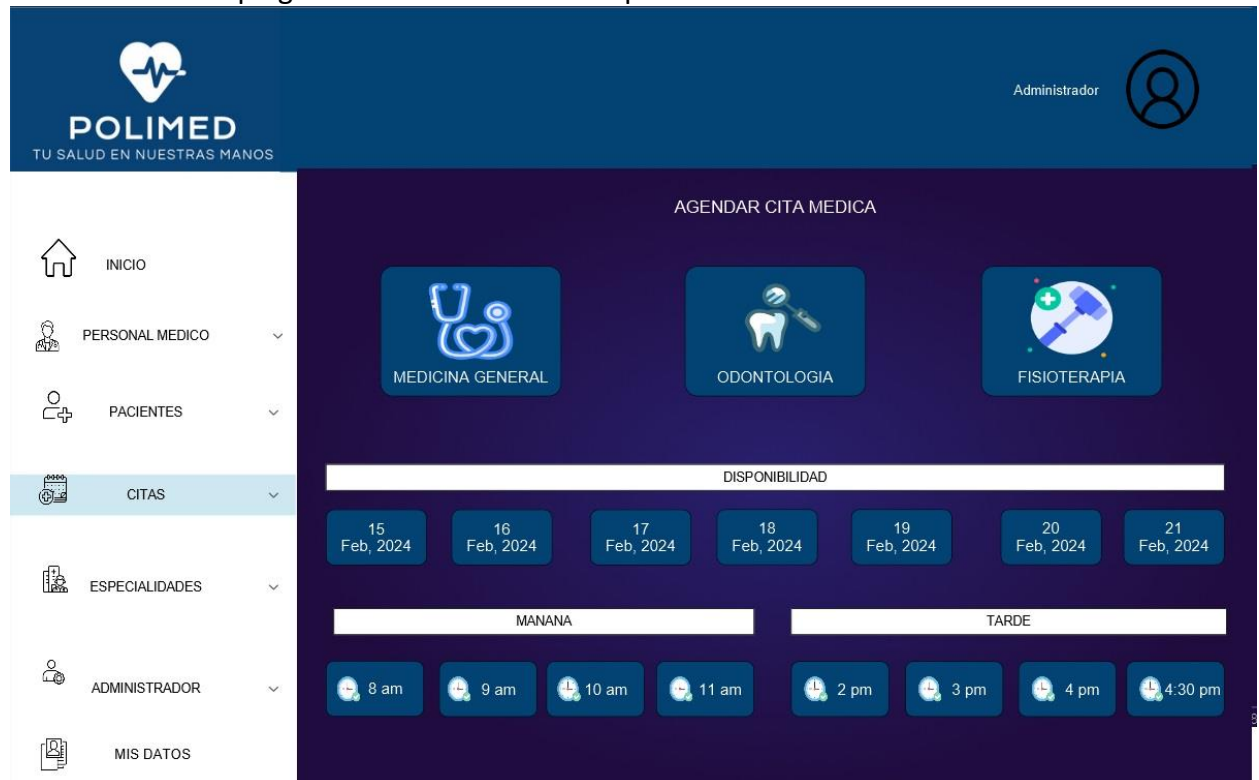
Alergia

Medicamento

 GUARDAR
  ELIMINAR
  ATRAS

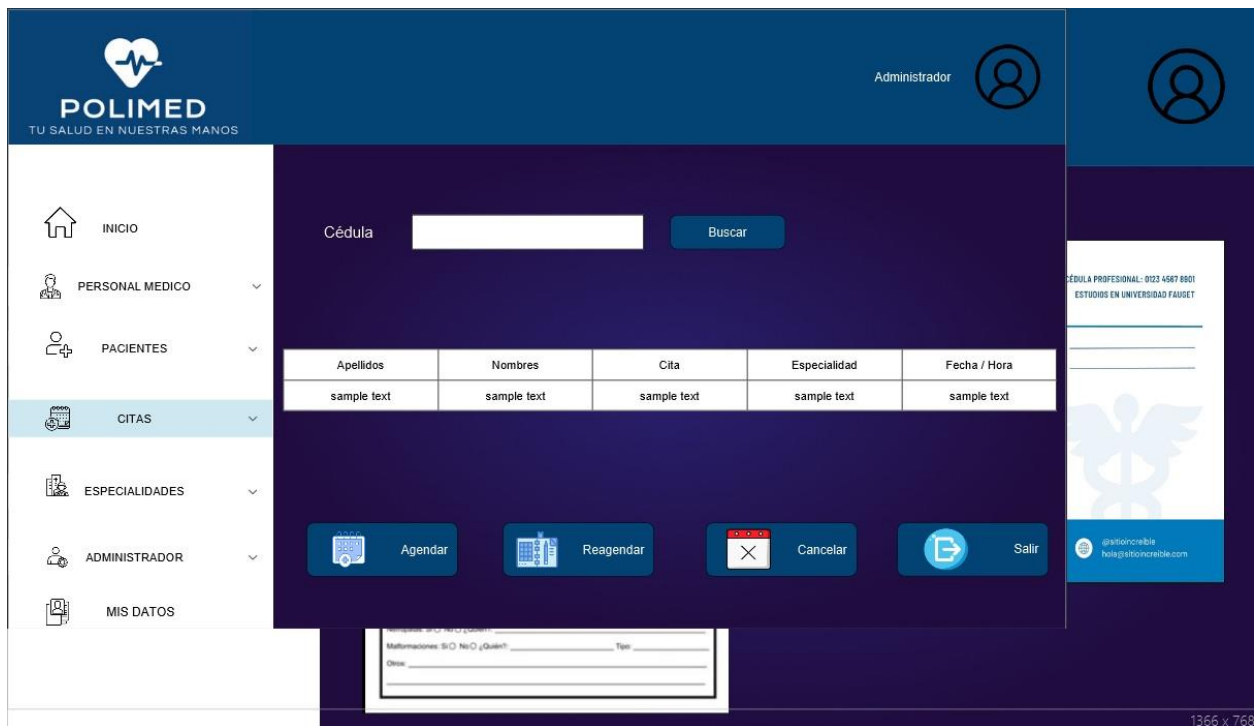
En la presente ventana, se puede gestionar los datos necesarios de los pacientes, contando con tres botones uno de guardar, el de eliminar para limpiar los datos ingresados, pero no guardados y un botón atrás que permitirá el retorno a la vista principal, es decir el inicio.

El botón citas desplegará dos vistas diferentes que son:



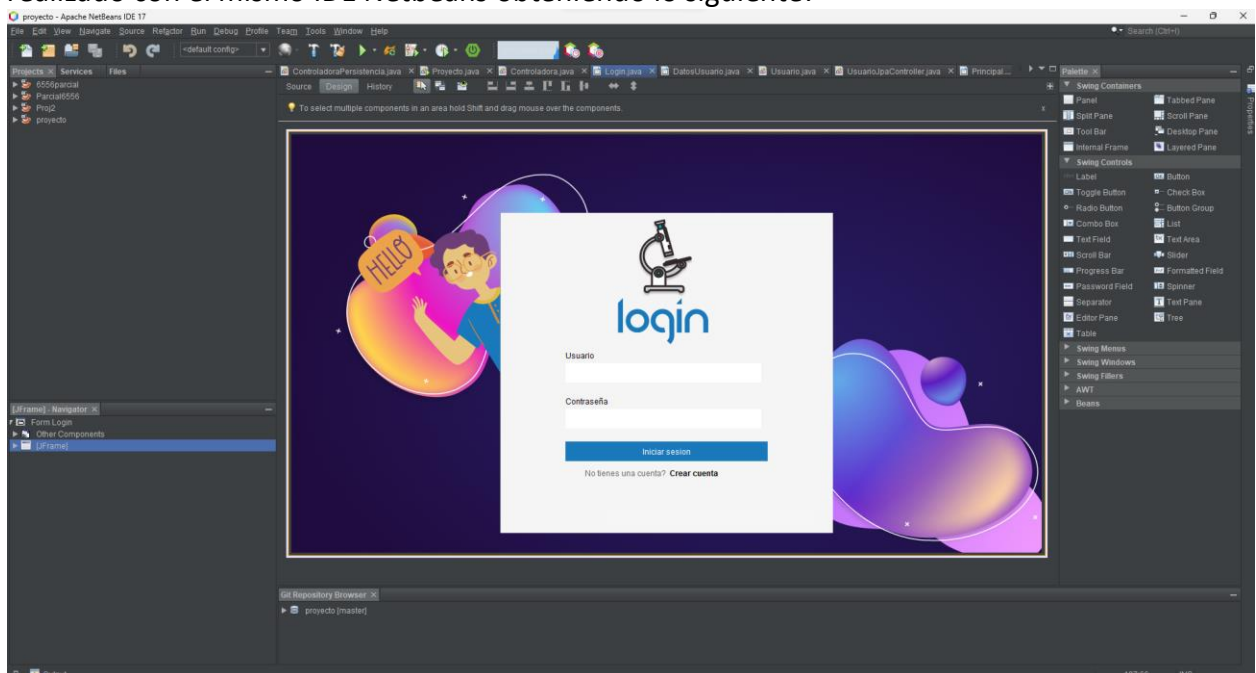
Desde esta ventana se podrá realizar el respectivo agendamiento de citas médicas, contando de arriba para abajo con la siguiente disposición:

1. Las especialidades: Medicina General, Odontología y Fisioterapia (Especialidades con las que cuenta el subcentro).
2. Los horarios disponibles para las citas medicas (las cuales serán desplegadas una vez se seleccione la especialidad).
3. Una vez seleccionados los dos apartados anteriores se desplegará los dos horarios de disponibilidad, tomando en cuenta el horario de atención del subcentro.
4. Una vez elegido el anterior apartado, se desplegará la disponibilidad de horario de atención médico, y de esta manera se realizará el agendamiento de la cita.



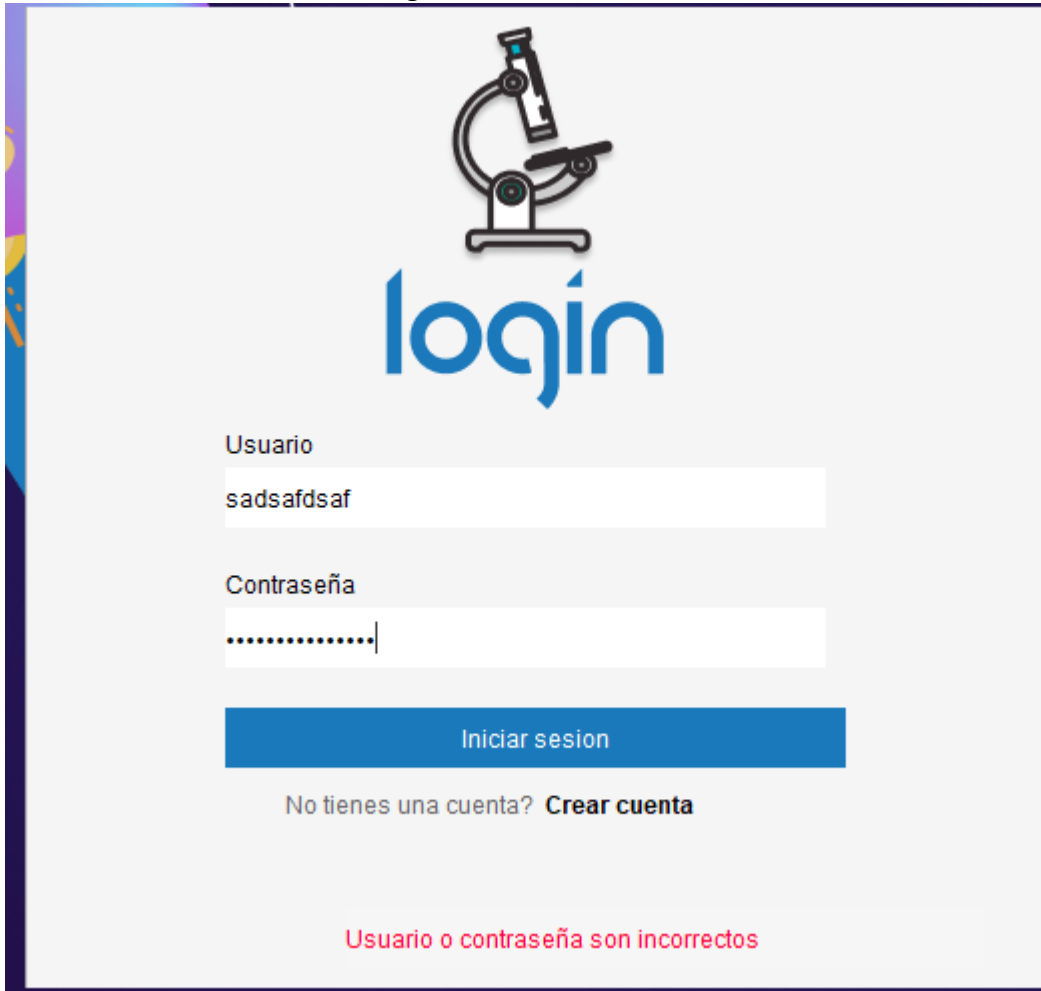
La ventana de gestión de citas, permitirá el observar y buscar el o los pacientes que se necesiten, por medio de su número de cedula, para que de esta manera la búsqueda se dé de la manera más óptima, desplegando la información del paciente, si tiene cita, en que especialidad y la fecha de la misma. En caso de que el paciente no se encuentre registrado, se procederá a su debido registro y se procederá al proceso de agendamiento de la cita médica que necesite. Además, desde la presente ventana se podrá cancelar, modificar o crear una cita médica.

Una vez realizados los mockups se ah procedido a crear las interfaces, en este caso se ha realizado con el mismo IDE Netbeans obteniendo lo siguiente:




La primera interfaz de usuario realizada fue, la de login la cual cuenta con ciertos controles de seguridad, pero ah petición del usuario no se ah impuesto un conteo de intentos máximos de ingreso.

Como se hablo anteriormente, se ah realizado el control del funcionamiento correcto de esta interfaz, teniendo como resultado lo siguiente:



The image shows a login interface with a light gray background. At the top center is a black and white icon of a microscope. Below the icon is the word "login" in a large, blue, sans-serif font. Underneath the logo are two input fields. The first is labeled "Usuario" and contains the text "sadsafdsaf". The second is labeled "Contraseña" and contains a series of dots, indicating a password. Below these fields is a blue button with the text "Iniciar sesion". Under the button is a link that says "No tienes una cuenta? **Crear cuenta**". At the bottom of the interface, there is a red error message that reads "Usuario o contraseña son incorrectos".

En caso de que tanto el usuario, como la contraseña, sean incorrectos, se mostrará el siguiente mensaje “**Usuario o contraseña son incorrectos**”.



login

Usuario

Contraseña

Iniciar sesion

No tienes una cuenta? [Crear cuenta](#)

Ingrese la contraseña correcta

En caso de que solo la contraseña sea incorrecta, se presentará el mensaje de “**Ingrese la contraseña correcta**”.

En caso de que se requiera crear una cuenta se desplegará la siguiente ventana:



REGISTRO

Nombre :

Apellido:

Email:

Username:

Contraseña:

GUARDAR CANCELAR

En la cual se procederá al ingreso de los datos del nuevo usuario y se dará click al botón guardar o en caso contrario el botón cancelar para volver a la ventana de login.

7. CONCLUSIONES Y RECOMENDACIONES:

- Como recomendación, para realizar los mockups, se debe utilizar tecnologías que permitan una buena experiencia y nos brinden no solo versatilidad, sino agilidad y simplicidad para optimizar el desarrollo del presente proceso, a su vez se debe complementar con teorías, como el caso de la teoría de colores, para brindar una buena experiencia visual al usuario y utilizar matices similares para mantener interfaces más coherentes.
- Como conclusión, se puede tomar en cuenta que la interfaz al tener no solo botones, sino también iconos que son familiares al usuario, sino también que cada ventana es intuitiva y se podría decir “se explica por si misma”, permitirá que el usuario se familiarice de una manera rápida al uso del sistema, a su vez que la información se muestra de una manera clara, con una interfaz coherente lo cual brinda una buena experiencia de uso para el usuario.

8. BIBLIOGRAFÍA:

- <https://concepto.de/teoria-del-color/>
- <https://dical.es/blog/disenio/en-que-consiste-la-teoria-del-color>
- <https://delaossa.co/teoria-del-color/>
- <https://ovacen.com/disenio-interfaz-usuario-ui/>
- <https://appmaster.io/es/blog/los-mejores-consejos-de-diseno-de-interfaz-de-usuario>
- <https://www.figma.com/files/recents-and-sharing?fuid=1324199383252128563>
- <https://www.justinmind.com>

ANEXOS (adjuntar el código fuente del programa/módulo desarrollado)

1. Interfaz de login

```
package com.mycompany.proyecto.igu;

import com.mycompany.proyecto.logica.Controladora;

public class Login extends javax.swing.JFrame {

    Controladora control;

    public Login() {

        initComponents();

        control = new Controladora();

    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jPanel2 = new javax.swing.JPanel();
        jLabel12 = new javax.swing.JLabel();
        BtnCuenta = new javax.swing.JButton();
        BtnIniciar = new javax.swing.JButton();
        txtUsuario = new javax.swing.JTextField();
        txtMensaje = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();

    }
}
```

```
txtContra = new javax.swing.JPasswordField();

jLabel11 = new javax.swing.JLabel();

jLabel1 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

jLabel7 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jLabel10 = new javax.swing.JLabel();

jLabel6 = new javax.swing.JLabel();

jLabel8 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);


jPanel1.setBackground(new java.awt.Color(255, 255, 255));

jPanel1.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());


jPanel2.setBackground(new java.awt.Color(204, 204, 204));

jPanel2.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());


jLabel12.setFont(new java.awt.Font("Leelawadee", 0, 12)); // NOI18N
jLabel12.setForeground(new java.awt.Color(102, 102, 102));
jLabel12.setText("No tienes una cuenta?");

jPanel2.add(jLabel12, new org.netbeans.lib.awtextra.AbsoluteConstraints(130, 390, -
1, -1));


BtnCuenta.setFont(new java.awt.Font("Leelawadee", 1, 12)); // NOI18N
BtnCuenta.setForeground(new java.awt.Color(0, 0, 0));
```

```
BtnCuenta.setText("Crear cuenta");

BtnCuenta.setBorder(null);

BtnCuenta.setBorderPainted(false);

BtnCuenta.setContentAreaFilled(false);

BtnCuenta.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        BtnCuentaActionPerformed(evt);

    }

});

jPanel2.add(BtnCuenta, new org.netbeans.lib.awtextra.AbsoluteConstraints(260,
390, -1, -1));
```

```
BtnIniciar.setBackground(new java.awt.Color(25, 121, 187));

BtnIniciar.setFont(new java.awt.Font("Leelawadee", 0, 12)); // NOI18N

BtnIniciar.setForeground(new java.awt.Color(255, 255, 255));

BtnIniciar.setText("Iniciar sesion");

BtnIniciar.setBorder(null);

BtnIniciar.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        BtnIniciarActionPerformed(evt);

    }

});

jPanel2.add(BtnIniciar, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 350,
310, 30));
```

```
txtUsuario.setBackground(new java.awt.Color(255, 255, 255));

txtUsuario.setForeground(new java.awt.Color(0, 0, 0));

txtUsuario.setBorder(null);
```

```
jPanel2.add(txtUsuario, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 230, 300, 30));
```

```
txtMensaje.setEditable(false);
```

```
txtMensaje.setBackground(new java.awt.Color(246, 246, 246));
```

```
txtMensaje.setForeground(new java.awt.Color(255, 0, 51));
```

```
txtMensaje.setBorder(null);
```

```
jPanel2.add(txtMensaje, new org.netbeans.lib.awtextra.AbsoluteConstraints(160, 450, 320, 30));
```

```
jPanel2.add(jTextField2, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 230, 300, 30));
```

```
txtContra.setBackground(new java.awt.Color(255, 255, 255));
```

```
txtContra.setForeground(new java.awt.Color(0, 0, 0));
```

```
txtContra.setBorder(null);
```

```
txtContra.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        txtContraActionPerformed(evt);
```

```
    }
```

```
});
```

```
jPanel2.add(txtContra, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 300, 300, 30));
```

```
jLabel11.setFont(new java.awt.Font("Leelawadee", 0, 12)); // NOI18N
```

```
jLabel11.setForeground(new java.awt.Color(0, 0, 0));
```

```
jLabel11.setText("Contraseña");
```

```
jPanel2.add(jLabel11, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 280, -1, -1));
```

```
jLabel1.setIcon(new  
javax.swing.ImageIcon("C:\\Users\\dalej\\OneDrive\\Documents\\NetBeansProjects\\pr  
oyecto\\src\\main\\java\\com\\mycompany\\proyecto\\igu\\img\\Group 41.png")); // NOI18N
```

```
jPanel2.add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(210, 10, -1, -  
1));
```

```
jLabel4.setIcon(new  
javax.swing.ImageIcon("C:\\Users\\dalej\\Downloads\\app\\Login.png")); // NOI18N
```

```
jPanel2.add(jLabel4, new org.netbeans.lib.awtextra.AbsoluteConstraints(180, 130, -1,  
-1));
```

```
jLabel7.setFont(new java.awt.Font("Leelawadee", 0, 12)); // NOI18N
```

```
jLabel7.setForeground(new java.awt.Color(0, 0, 0));
```

```
jLabel7.setText("Usuario");
```

```
jPanel2.add(jLabel7, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 210, -1,  
-1));
```

```
jLabel3.setIcon(new  
javax.swing.ImageIcon("C:\\Users\\dalej\\OneDrive\\Documents\\NetBeansProjects\\pr  
oyecto\\src\\main\\java\\com\\mycompany\\proyecto\\igu\\img\\Rectangle 1.png"));  
// NOI18N
```

```
jLabel3.setText("jLabel3");
```

```
jPanel2.add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 510,  
490));
```

```
jLabel10.setFont(new java.awt.Font("Leelawadee", 0, 14)); // NOI18N
```

```
jLabel10.setForeground(new java.awt.Color(0, 0, 0));
```

```
jLabel10.setText("Usuario");
```

```
jPanel2.add(jLabel10, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 210, -  
1, -1));
```

```
jPanel1.add(jPanel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(320, 120, 510, 490));
```

```
jLabel6.setIcon(new  
javax.swing.ImageIcon("C:\\Users\\dalej\\OneDrive\\Documents\\NetBeansProjects\\pr  
oyecto\\src\\main\\java\\com\\mycompany\\proyecto\\igu\\img\\form 2.png")); //  
NOI18N
```

```
jLabel6.setText("jLabel6");
```

```
jPanel1.add(jLabel6, new org.netbeans.lib.awtextra.AbsoluteConstraints(780, 310, -1, -1));
```

```
jLabel8.setIcon(new  
javax.swing.ImageIcon("C:\\Users\\dalej\\OneDrive\\Documents\\NetBeansProjects\\pr  
oyecto\\src\\main\\java\\com\\mycompany\\proyecto\\igu\\img\\3161253 1.png")); //  
NOI18N
```

```
jPanel1.add(jLabel8, new org.netbeans.lib.awtextra.AbsoluteConstraints(90, 70, -1, 400));
```

```
jLabel5.setIcon(new  
javax.swing.ImageIcon("C:\\Users\\dalej\\OneDrive\\Documents\\NetBeansProjects\\pr  
oyecto\\src\\main\\java\\com\\mycompany\\proyecto\\igu\\img\\form 1.png")); //  
NOI18N
```

```
jLabel5.setText("jLabel5");
```

```
jPanel1.add(jLabel5, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, -1, -1));
```

```
jLabel2.setIcon(new  
javax.swing.ImageIcon("C:\\Users\\dalej\\OneDrive\\Documents\\NetBeansProjects\\pr  
oyecto\\src\\main\\java\\com\\mycompany\\proyecto\\igu\\img\\Frame 3.jpg")); //  
NOI18N
```

```
jPanel1.add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(-10, -60, 1390, 850));
```

```
javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
```

```
getContentPane().setLayout(layout);
```

```

        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGap(0, 1159, Short.MAX_VALUE)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(0, 4, Short.MAX_VALUE)

                .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, 1150,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 5, Short.MAX_VALUE)))
        );

        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGap(0, 648, Short.MAX_VALUE)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(0, 4, Short.MAX_VALUE)

                .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE, 640,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 4, Short.MAX_VALUE)))
        );

        pack();
    } // </editor-fold>

```

```

private void BtnCuentaActionPerformed(java.awt.event.ActionEvent evt) {
    DatosUsuario datusu= new DatosUsuario();

    datusu.setVisible(true);

    datusu.setLocationRelativeTo(null);
}

```



```
}
```

```
private void BtnIniciarActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    String user = txtUsuario.getText();
```

```
    String contra = txtContra.getText();
```

```
    String mensaje = control.validarUsuario(user,contra);
```

```
    txtMensaje.setText(mensaje);
```

```
    txtUsuario.setText("");
```

```
    txtContra.setText("");
```

```
    Principal prin = new Principal();
```

```
    prin.setVisible(true);
```

```
}
```

```
private void txtContraActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    // TODO add your handling code here:
```

```
}
```

```
// Variables declaration - do not modify
```

```
private javax.swing.JButton BtnCuenta;
```

```
private javax.swing.JButton BtnIniciar;
```

```
private javax.swing.JLabel jLabel1;
```

```
private javax.swing.JLabel jLabel10;
```

```
private javax.swing.JLabel jLabel11;
```

```
private javax.swing.JLabel jLabel12;
```

```
private javax.swing.JLabel jLabel2;
```

```

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JTextField jTextField2;

private javax.swing.JPasswordField txtContra;

private javax.swing.JTextField txtMensaje;

private javax.swing.JTextField txtUsuario;

// End of variables declaration
}

```

2. Interfaz de creación de nueva cuenta

```

package com.mycompany.proyecto.igu;

import com.mycompany.proyecto.logica.Controladora;

import com.mycompany.proyecto.logica.Usuario;

import java.sql.Date;

import java.util.Calendar;

public class DatosUsuario extends javax.swing.JFrame {

    Controladora control;

    public DatosUsuario() {

        initComponents();
    }
}

```

```
control = new Controladora();  
}
```

```
@SuppressWarnings("unchecked")
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
private void initComponents() {
```

```
    jPanel1 = new javax.swing.JPanel();
```

```
    jLabel1 = new javax.swing.JLabel();
```

```
    jLabel3 = new javax.swing.JLabel();
```

```
    jLabel4 = new javax.swing.JLabel();
```

```
    jLabel5 = new javax.swing.JLabel();
```

```
    jLabel6 = new javax.swing.JLabel();
```

```
    jLabel7 = new javax.swing.JLabel();
```

```
    btnGuardarUsu = new javax.swing.JButton();
```

```
    btnCancelarUsu = new javax.swing.JButton();
```

```
    txtNombreUsuario = new javax.swing.JTextField();
```

```
    txtApeUsuario = new javax.swing.JTextField();
```

```
    txtEmailUsuario = new javax.swing.JTextField();
```

```
    txtAliasUsuario = new javax.swing.JTextField();
```

```
    txtContraUsuario = new javax.swing.JTextField();
```

```
    jLabel2 = new javax.swing.JLabel();
```

```
    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
    jPanel1.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());
```

```
jLabel1.setFont(new java.awt.Font("Gloucester MT Extra Condensed", 1, 48)); // NOI18N
```

```
jLabel1.setForeground(new java.awt.Color(255, 255, 255));
```

```
jLabel1.setText("REGISTRO");
```

```
jPanel1.add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(110, 40, -1, -1));
```

```
jLabel3.setFont(new java.awt.Font("Dialog", 1, 24)); // NOI18N
```

```
jLabel3.setForeground(new java.awt.Color(255, 255, 255));
```

```
jLabel3.setText("Nombre :");
```

```
jPanel1.add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 140, -1, -1));
```

```
jLabel4.setFont(new java.awt.Font("Dialog", 1, 24)); // NOI18N
```

```
jLabel4.setForeground(new java.awt.Color(255, 255, 255));
```

```
jLabel4.setText("Apellido:");
```

```
jPanel1.add(jLabel4, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 190, -1, -1));
```

```
jLabel5.setFont(new java.awt.Font("Dialog", 1, 24)); // NOI18N
```

```
jLabel5.setForeground(new java.awt.Color(255, 255, 255));
```

```
jLabel5.setText("Email:");
```

```
jPanel1.add(jLabel5, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 240, -1, -1));
```

```
jLabel6.setFont(new java.awt.Font("Dialog", 1, 24)); // NOI18N
```

```
jLabel6.setForeground(new java.awt.Color(255, 255, 255));
```

```
jLabel6.setText("Username:");
```

```
jPanel1.add(jLabel6, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 290, -1, -1));
```

```
jLabel7.setFont(new java.awt.Font("Dialog", 1, 24)); // NOI18N
jLabel7.setForeground(new java.awt.Color(255, 255, 255));
jLabel7.setText("Contraseña:");
jPanel1.add(jLabel7, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 340, -1, -1));
```

```
btnGuardarUsu.setBackground(new java.awt.Color(255, 255, 255));
btnGuardarUsu.setForeground(new java.awt.Color(0, 0, 0));
btnGuardarUsu.setText("GUARDAR");
btnGuardarUsu.setBorder(null);
btnGuardarUsu.setBorderPainted(false);
btnGuardarUsu.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnGuardarUsuActionPerformed(evt);
    }
});
jPanel1.add(btnGuardarUsu, new org.netbeans.lib.awtextra.AbsoluteConstraints(60, 450, 120, 50));
```

```
btnCancelarUsu.setBackground(new java.awt.Color(255, 255, 255));
btnCancelarUsu.setForeground(new java.awt.Color(0, 0, 0));
btnCancelarUsu.setText("CANCELAR");
btnCancelarUsu.setBorder(null);
btnCancelarUsu.setBorderPainted(false);
btnCancelarUsu.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnCancelarUsuActionPerformed(evt);
    }
});
```

```

});

jPanel1.add(btnCancelarUsu, new
org.netbeans.lib.awtextra.AbsoluteConstraints(200, 450, 120, 50));

txtNombreUsuario.setFont(new java.awt.Font("Dialog", 0, 14)); // NOI18N
txtNombreUsuario.setBorder(null);

txtNombreUsuario.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        txtNombreUsuarioActionPerformed(evt);

    }

});

jPanel1.add(txtNombreUsuario, new
org.netbeans.lib.awtextra.AbsoluteConstraints(160, 140, 210, 30));

txtApeUsuario.setFont(new java.awt.Font("Dialog", 0, 14)); // NOI18N
txtApeUsuario.setBorder(null);

jPanel1.add(txtApeUsuario, new org.netbeans.lib.awtextra.AbsoluteConstraints(160,
190, 210, 30));

txtEmailUsuario.setFont(new java.awt.Font("Dialog", 0, 14)); // NOI18N
txtEmailUsuario.setBorder(null);

jPanel1.add(txtEmailUsuario, new
org.netbeans.lib.awtextra.AbsoluteConstraints(160, 240, 210, 30));

txtAliasUsuario.setFont(new java.awt.Font("Dialog", 0, 14)); // NOI18N
txtAliasUsuario.setBorder(null);

jPanel1.add(txtAliasUsuario, new org.netbeans.lib.awtextra.AbsoluteConstraints(160,
290, 210, 30));

txtContraUsuario.setFont(new java.awt.Font("Dialog", 0, 14)); // NOI18N

```

```

        txtContraUsuario.setBorder(null);

        jPanel1.add(txtContraUsuario, new
org.netbeans.lib.awtextra.AbsoluteConstraints(160, 340, 210, 30));

        jLabel2.setIcon(new
javax.swing.ImageIcon("C:\\Users\\dalej\\OneDrive\\Documents\\NetBeansProjects\\pr
oyecto\\src\\main\\java\\com\\mycompany\\proyecto\\igu\\img\\Frame 3.jpg")); //
NOI18N

        jPanel1.add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 380,
567));

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);

        layout.setHorizontalGroup(

            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        );

        layout.setVerticalGroup(

            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

        );

        pack();

    } // </editor-fold>

    private void txtNombreUsuarioActionPerformed(java.awt.event.ActionEvent evt) {

    }

```

```

private void btnGuardarUsuActionPerformed(java.awt.event.ActionEvent evt) {

    // Crear un objeto de tipo Calendar para obtener la fecha actual

    Calendar calendar = Calendar.getInstance();

    // Obtener la fecha actual

    java.util.Date fechaActual = calendar.getTime();

    // Dar valores de fecha a fechanacimiento y fecharegistro

    java.util.Date fechaRegistro = fechaActual; // Fecha actual como fecha de registro

    String nombusu = txtNombreUsuario.getText();

    String apeusu = txtApeUsuario.getText();

    String emailusu = txtEmailUsuario.getText();

    String aliasu = txtAliasUsuario.getText();

    String conusu = txtContraUsuario.getText();

    Usuario nuevoUsuario = new Usuario(Long.MIN_VALUE, aliasu, conusu, nombusu,
apeusu, emailusu, fechaRegistro);

    control.crearUsuario(nuevoUsuario);

    txtNombreUsuario.setText("");

    txtApeUsuario.setText("");

    txtEmailUsuario.setText("");

    txtAliasUsuario.setText("");

    txtContraUsuario.setText("");

}

private void btnCancelarUsuActionPerformed(java.awt.event.ActionEvent evt) {

    dispose();

}

```



```

// Variables declaration - do not modify

private javax.swing.JButton btnCancelarUsu;

private javax.swing.JButton btnGuardarUsu;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JPanel jPanel1;

private javax.swing.JTextField txtAliasUsuario;

private javax.swing.JTextField txtApeUsuario;

private javax.swing.JTextField txtContraUsuario;

private javax.swing.JTextField txtEmailUsuario;

private javax.swing.JTextField txtNombreUsuario;

// End of variables declaration
}

```

3. Código de la persistencia

```

<?xml version="1.0" encoding="UTF-8"?>

<persistence version="2.1" 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">

<persistence-unit name="proJPAPU" transaction-type="RESOURCE_LOCAL">

    <provider>org.eclipse.persistence.jpa.PersistenceProvider</provider>

    <class>com.mycompany.proyecto.logica.Paciente</class>

    <class>com.mycompany.proyecto.logica.Usuario</class>

    <class>com.mycompany.proyecto.logica.Especialidad</class>

    <class>com.mycompany.proyecto.logica.Medico</class>

    <class>com.mycompany.proyecto.logica.Reservacion</class>

    <class>com.mycompany.proyecto.logica.Estado</class>

    <properties>

        <property name="javax.persistence.jdbc.url"
value="jdbc:mysql://localhost:3306/polimed?serverTimezone=UTC"/>

        <property name="javax.persistence.jdbc.user" value="root"/>

        <property name="javax.persistence.jdbc.driver" value="com.mysql.cj.jdbc.Driver"/>

        <property name="javax.persistence.jdbc.password" value=""/>

        <property name="javax.persistence.schema-generation.database.action"
value="create"/>

    </properties>

</persistence-unit>

</persistence>

```

4. Código de controladora de persistencia

```

package com.mycompany.proyecto.persistencia;

import com.mycompany.proyecto.logica.Especialidad;

import com.mycompany.proyecto.logica.Estado;

import com.mycompany.proyecto.logica.Medico;

import com.mycompany.proyecto.logica.Paciente;

import com.mycompany.proyecto.logica.Reservacion;

```

```
import com.mycompany.proyecto.logica.Usuario;

import com.mycompany.proyecto.persistencia.exceptions.NonexistentEntityException;

import java.util.ArrayList;

import java.util.List;

import java.util.logging.Level;

import java.util.logging.Logger;


public class ControladoraPersistencia {

    PacienteJpaController pacJpa = new PacienteJpaController();

    UsuarioJpaController usuJpa = new UsuarioJpaController();

    EspecialidadJpaController espJpa = new EspecialidadJpaController();

    EstadoJpaController estJpa = new EstadoJpaController();

    MedicoJpaController medJpa = new MedicoJpaController();

    ReservacionJpaController resJpa = new ReservacionJpaController();


    //Paciente

    public void crearPaciente(Paciente pac) {

        pacJpa.create(pac);

    }


    public void eliminarPaciente(long id) {

        try {

            pacJpa.destroy(id);

        } catch (NonexistentEntityException ex) {

            Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE, null, ex);

        }

    }

}
```

```

    }

    public void editarPaciente(Paciente pac){

        try{

            pacJpa.edit(pac);

        }catch(Exception ex){

            Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE,null,ex);

        }

    }

    public List<Paciente> traerPacientes() {

        return pacJpa.findPacienteEntities();

    }

    /* public Paciente traerPaciente(long id) {

        return pacJpa.findPaciente(id);

    }

    public ArrayList<Paciente> traerListaPacientes() {

        List<Paciente> lista = pacJpa.findPacienteEntities();

        ArrayList<Paciente> listaPacientes = new ArrayList<Paciente>(lista);

        return listaPacientes;

    }*/

    //Usuario

    public void crearUsuario(Usuario usu) {

        usuJpa.create(usu);

```

```
}
```

```
public void eliminarUsuario(long id) {
```

```
    try {
```

```
        usuJpa.destroy(id);
```

```
    } catch (NonexistentEntityException ex) {
```

```
        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE, null, ex);
```

```
    }
```

```
}
```

```
public void editarUsuario(Usuario usu) {
```

```
    try{
```

```
        usuJpa.edit(usu);
```

```
    }catch(Exception ex){
```

```
        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE,null,ex);
```

```
    }
```

```
}
```

```
public List<Usuario> traerUsuarios() {
```

```
    List<Usuario>listaUsuarios= usuJpa.findUsuarioEntities();
```

```
    return listaUsuarios;
```

```
}
```

```
/* public ArrayList<Usuario> traerListaUsuarios() {
```

```
    List<Usuario> listaus = usuJpa.findUsuarioEntities();
```

```
    ArrayList<Usuario> listaUsuarios = new ArrayList<Usuario>(listaus);
```

```
    return listaUsuarios;
```

```
*/
```

```
//Especialidad
```

```

public void crearEspecialidad(Especialidad esp) {
    espJpa.create(esp);
}

public void eliminarEspecialidad(long id) {
    try {
        espJpa.destroy(id);
    } catch (NonexistentEntityException ex) {
        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE, null, ex);
    }
}

public void editarEspecialidad(Especialidad esp) {
    try{
        espJpa.edit(esp);
    }catch(Exception ex){
        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE,null,ex);
    }
}

public Especialidad traerEspecialidad(long id) {
    return espJpa.findEspecialidad(id);
}

public ArrayList<Especialidad> traerListaEspecialidades() {

    List<Especialidad> listaesp = espJpa.findEspecialidadEntities();

    ArrayList<Especialidad> listaEspecialidades = new ArrayList<Especialidad>(listaesp);

    return listaEspecialidades;
}

```

```
//Medico

public void crearMedico(Medico med) {

    medJpa.create(med);

}


public void eliminarMedico(long id) {

    try {

        medJpa.destroy(id);

    } catch (NonexistentEntityException ex) {

        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE, null, ex);

    }

}


public void editarMedico(Medico med) {

    try{

        medJpa.edit(med);

    }catch(Exception ex){

        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE,null,ex);

    }

}


public List<Medico> traerMedicos() {

    return medJpa.findMedicoEntities();

}


/* public Medico traerMedico(long id) {

    return medJpa.findMedico(id);

}

public ArrayList<Medico> traerListaMedicos() {
```

```

        List<Medico> listamed = medJpa.findMedicoEntities();

        ArrayList<Medico> listaMedicos = new ArrayList<Medico>(listamed);

        return listaMedicos;
    }*/

    //Reservacion

    public void crearReservacion(Reservacion res) {

        resJpa.create(res);
    }

    public void eliminarReservacion(long id) {

        try {

            resJpa.destroy(id);

        } catch (NonexistentEntityException ex) {

            Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE, null, ex);

        }

    }

    public void editarReservacion(Reservacion res) {

        try{

            resJpa.edit(res);

        }catch(Exception ex){

            Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE,null,ex);

        }

    }

    public Reservacion traerReservacion(long id) {

        return resJpa.findReservacion(id);
    }

```



```

}

public ArrayList<Reservacion> traerListaReservaciones() {

    List<Reservacion> listares = resJpa.findReservacionEntities();

    ArrayList<Reservacion> listaReservaciones = new ArrayList<Reservacion>(listares);

    return listaReservaciones;
}

// Estado

public void crearEstado(Estado estado) {

    estJpa.create(estado);
}

public void eliminarEstado(long id) {

    try {

        estJpa.destroy(id);

    } catch (NonexistentEntityException ex) {

        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE, null, ex);

    }

}

public void editarEstado(Estado estado) {

    try {

        estJpa.edit(estado);

    } catch (Exception ex) {

        Logger.getLogger(ControladoraPersistencia.class.getName()).log(Level.SEVERE, null, ex);

    }

}

```

```

public Estado traerEstado(long id) {
    return estJpa.findEstado(id);
}

public ArrayList<Estado> traerListaEstados() {
    List<Estado> listaEstados = estJpa.findEstadoEntities();
    return new ArrayList<>(listaEstados);
}

public List<Estado> obtenerTiposDeEstados() {
    return estJpa.findEstadoEntities();
}

}

```

5. Código de controladoras

```

package com.mycompany.proyecto.logica;

import com.mycompany.proyecto.persistencia.ControladoraPersistencia;
import java.util.List;

public class Controladora {

    ControladoraPersistencia controlPersis;

```

```
public Controladora() {  
    controlPersis= new ControladoraPersistencia();  
}
```

```
//Paciente
```

```
public void crearPaciente(Paciente pac){  
    controlPersis.crearPaciente(pac);  
}
```

```
public void eliminarPaciente(int id){  
    controlPersis.eliminarPaciente(id);  
}
```

```
public void editarPaciente(Paciente pac){  
    controlPersis.editarPaciente(pac);  
}
```

```
/*public Paciente traerPaciente(int id){  
    return controlPersis.traerPaciente(id);  
}
```

```
public ArrayList<Paciente> traerListaPacientes(Paciente pac){  
    List<Paciente> lista = controlPersis.traerListaPacientes();  
    ArrayList<Paciente> listaPacientes = new ArrayList<>(lista);  
    return listaPacientes;
```

```
// return controlPersis.traerListaPacientes();  
}*/
```

```
//Usuario
```

```
public void crearUsuario(Usuario usu){
```

```

controlPersis.crearUsuario(usu);

}

public void eliminarUsuario(int id){

    controlPersis.eliminarUsuario(id);

}

public void editarUsuario(Usuario usu){

    controlPersis.editarUsuario(usu);

}

//public Usuario traerUsuario(int id){

//return controlPersis.traerUsuario(id);

//}

/*public ArrayList<Usuario> traerListaUsuarios(){

    List<Usuario> lista = controlPersis.traerListaUsuarios();

    ArrayList<Usuario> listaUsuarios = new ArrayList<>(lista);

    return listaUsuarios;

}

//return controlPersis.traerListaUsuarios();

}*/

public String validarUsuario(String user, String contra) {

    String mensaje="";

    List<Usuario> listaUsuarios = controlPersis.traerUsuarios();

    for(Usuario usu: listaUsuarios){

        if(usu.getUsername().equals(user)){

            if(usu.getPassword().equals(contra)){

                mensaje="";

                return mensaje;

            }

        }

    }

    return mensaje;
}

```

```
    }else{  
        mensaje="Ingrese la contraseña correcta";  
        return mensaje;  
    }  
    }else{  
        mensaje ="Usuario o contraseña son incorrectos";  
        return mensaje;  
    }  
    }return mensaje;  
}
```

//Especialidad

```
public void crearEspecialidad(Especialidad esp){  
    controlPersis.crearEspecialidad(esp);  
}  
  
public void eliminarEspecialidad(int id){  
    controlPersis.eliminarEspecialidad(id);  
}  
  
public void editarEspecialidad(Especialidad esp){  
    controlPersis.editarEspecialidad(esp);  
}  
  
/* public Especialidad traerEspecialidad(int id){  
    return controlPersis.traerEspecialidad(id);  
}  
  
public ArrayList<Especialidad> traerListaEspecialidades(){  
    List<Especialidad> lista = controlPersis.traerListaEspecialidades();
```

```

        ArrayList<Especialidad> listaEspecialidades = new ArrayList<>(lista);

        return listaEspecialidades;

//return controlPersis.traerListaEspecialidades();
    }*/

    //Medico

    public void crearMedico(Medico med){

        controlPersis.crearMedico(med);

    }

    public void eliminarMedico(int id){

        controlPersis.eliminarMedico(id);

    }

    public void editarMedico(Medico med){

        controlPersis.editarMedico(med);

    }

    /*public Medico traerMedico(int id){

        return controlPersis.traerMedico(id);

    }

    public ArrayList<Medico> traerListaMedicos(){

        List<Medico> lista = controlPersis.traerListaMedicos();

        ArrayList<Medico> listaMedicos = new ArrayList<>(lista);

        return listaMedicos;

//return controlPersis.traerListaMedicos();

    }*/

    //Reservacion

    public void crearReservacion(Reservacion res){

        controlPersis.crearReservacion(res);

```

```

    }

    public void eliminarReservacion(int id){
        controlPersis.eliminarReservacion(id);
    }

    public void editarReservacion(Reservacion res){
        controlPersis.editarReservacion(res);
    }

    /*public Reservacion traerReservacion(int id){
        return controlPersis.traerReservacion(id);
    }

    public ArrayList<Reservacion> traerListaReservaciones(){
        List<Reservacion> lista = controlPersis.traerListaReservaciones();
        ArrayList<Reservacion> listaReservaciones = new ArrayList<>(lista);
        return listaReservaciones;
    }
    //return controlPersis.traerListaReservaciones();
    */

    // Estado

    public void crearEstado(Estado estado) {
        controlPersis.crearEstado(estado);
    }

    public void eliminarEstado(long id) {
        controlPersis.eliminarEstado(id);
    }

    public void editarEstado(Estado estado) {
        controlPersis.editarEstado(estado);
    }

```

```

/* public Estado traerEstado(long id) {

    return controlPersis.traerEstado(id);

}

public ArrayList<Estado> traerListaEstados() {

    List<Estado> lista = controlPersis.traerListaEstados();

    ArrayList<Estado> listaEstados = new ArrayList<>(lista);

    return listaEstados;

    //return controlPersis.traerListaEstados();

}

public void mostrarTiposDeEstados() {

    List<Estado> tiposDeEstados = obtenerTiposDeEstados();

    System.out.println("Tipos de Estados:");

    for (Estado estado : tiposDeEstados) {

        System.out.println("ID: " + estado.getId() + ", Tipo: " + estado.getTipo());

    }

}

public List<Estado> obtenerTiposDeEstados() {

    return controlPersis.findEstadoEntities();

}*/

}

```


6. Código de entidades

- Usuario

```
package com.mycompany.proyecto.logica;

import java.io.Serializable;

import java.util.Date;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

import javax.persistence.Temporal;

import javax.persistence.TemporalType;

@Entity

@Table(name="usuario")

public class Usuario implements Serializable {

    @Id

    // @GeneratedValue(strategy=GenerationType.AUTO)

    //private int id;

    @GeneratedValue(strategy = GenerationType.IDENTITY)

    private Long id;

    private String username;

    private String password;

    private String nombre_user;

    private String apellido_usuario;

    @Column(unique=true)
```

```
private String email_user;

private Boolean activo=true;

private Boolean admi=false;

@Temporal(TemporalType.TIMESTAMP)

private java.util.Date fecharegistro;
```

```
public Usuario() {

}
```

```
/* public Usuario(int id, String username, String password, String nombre_user,
String apellido_usuario, String email_user, Boolean activo, Boolean admi, Date
fecharegistro) {
```

```
    this.id = id;

    this.username = username;

    this.password = password;

    this.nombre_user = nombre_user;

    this.apellido_usuario = apellido_usuario;

    this.email_user = email_user;

    this.activo = activo;

    this.admi = admi;

    this.fecharegistro = fecharegistro;

}
```

```
public int getId() {

    return id;

}
```

```
public void setId(int id) {  
    this.id = id;  
}
```

```
public String getUsername() {  
    return username;  
}
```

```
public void setUsername(String username) {  
    this.username = username;  
}
```

```
public String getPassword() {  
    return password;  
}
```

```
public void setPassword(String password) {  
    this.password = password;  
}
```

```
public String getNombre_user() {  
    return nombre_user;  
}
```

```
public void setNombre_user(String nombre_user) {  
    this.nombre_user = nombre_user;  
}
```

```
public String getApellido_usuario() {  
    return apellido_usuario;  
}
```

```
public void setApellido_usuario(String apellido_usuario) {  
    this.apellido_usuario = apellido_usuario;  
}
```

```
public String getEmail_user() {  
    return email_user;  
}
```

```
public void setEmail_user(String email_user) {  
    this.email_user = email_user;  
}
```

```
public Boolean getActivo() {  
    return activo;  
}
```

```
public void setActivo(Boolean activo) {  
    this.activo = activo;  
}
```

```
public Boolean getAdmi() {  
    return admi;  
}
```

```
public void setAdmi(Boolean admi) {  
  
    this.admi = admi;  
  
}
```

```
public Date getFecharregistro() {  
  
    return fecharregistro;  
  
}
```

```
public void setFecharregistro(Date fecharregistro) {  
  
    this.fecharregistro = fecharregistro;  
  
}
```

@Override

```
public String toString() {  
  
    return "Usuario{" + "id=" + id + ", username=" + username + ", password=" +  
password + ", nombre_user=" + nombre_user + ", apellido_usuario=" +  
apellido_usuario + ", email_user=" + email_user + ", activo=" + activo + ", admi=" +  
admi + ", fecharregistro=" + fecharregistro + '}';  
  
}*/
```

```
public Usuario(Long id, String username, String password, String nombre_user,  
String apellido_usuario, String email_user, Date fecharregistro) {  
  
    this.id = id;  
  
    this.username = username;  
  
    this.password = password;  
  
    this.nombre_user = nombre_user;  
  
    this.apellido_usuario = apellido_usuario;  
  
    this.email_user = email_user;  
  
    this.fecharregistro = fecharregistro;  
  
}
```

```
public Long getId() {  
    return id;  
}
```

```
public void setId(Long id) {  
    this.id = id;  
}
```

```
public String getUsername() {  
    return username;  
}
```

```
public void setUsername(String username) {  
    this.username = username;  
}
```

```
public String getPassword() {  
    return password;  
}
```

```
public void setPassword(String password) {  
    this.password = password;  
}
```

```
public String getNombre_user() {  
    return nombre_user;  
}
```

```
public void setNombre_user(String nombre_user) {  
    this.nombre_user = nombre_user;  
}
```

```
public String getApellido_usuario() {  
    return apellido_usuario;  
}
```

```
public void setApellido_usuario(String apellido_usuario) {  
    this.apellido_usuario = apellido_usuario;  
}
```

```
public String getEmail_user() {  
    return email_user;  
}
```

```
public void setEmail_user(String email_user) {  
    this.email_user = email_user;  
}
```

```
public Boolean getActivo() {  
    return activo;  
}
```

```
public void setActivo(Boolean activo) {  
    this.activo = activo;  
}
```

```

public Boolean getAdmi() {

    return admi;

}

```

```

public void setAdmi(Boolean admi) {

    this.admi = admi;

}

```

```

public java.util.Date getFecharegistro() {

    return fecharegistro;

}

```

```

public void setFecharegistro(java.util.Date fecharegistro) {

    this.fecharegistro = fecharegistro;

}

```

@Override

```

public String toString() {

    return "Usuario{" + "id=" + id + ", username=" + username + ", password=" +
password + ", nombre_user=" + nombre_user + ", apellido_usuario=" +
apellido_usuario + ", email_user=" + email_user + ", activo=" + activo + ", admi=" +
admi + ", fecharegistro=" + fecharegistro + '}';

}

```

```

}

```

- Medico


```
package com.mycompany.proyecto.logica;
```

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

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

```
import javax.persistence.Basic;
```

```
import javax.persistence.Column;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.JoinColumn;
```

```
import javax.persistence.ManyToOne;
```

```
import javax.persistence.Table;
```

```
import javax.persistence.Temporal;
```

```
import javax.persistence.TemporalType;
```

```
@Entity
```

```
@Table(name="medico")
```

```
public class Medico implements Serializable {
```

```
    @Id
```

```
    //@GeneratedValue(strategy=GenerationType.SEQUENCE)
```

```
    //private int id;
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    @Basic
```

```
    private String nombre_med;
```

```
    private String apellido_med;
```

```
    private String genero_med;
```

```
@Column(unique=true)

private String email_med;

private String direccion_med;

private String telefono;

private Boolean activo=true;

@Temporal(TemporalType.DATE)

private Date fechanacimiento;

@Temporal(TemporalType.TIMESTAMP)

private java.util.Date fecharegistro;
```

```
@ManyToOne

@JoinColumn(name="especialidad_id")

private Especialidad esp;
```

```
/*@OneToMany(mappedBy="med")

private LinkedList<Reservacion> listaReservaciones;*/

public Medico() {

}
```

```
/* public Medico(int id, String nombre_med, String apellido_med, String
genero_med, String email_med, String direccion_med, String telefono, Boolean activo,
Date fechanacimiento, Date fecharegistro, Especialidad esp, LinkedList<Reservacion>
listaReservaciones) {

    this.id = id;

    this.nombre_med = nombre_med;

    this.apellido_med = apellido_med;

    this.genero_med = genero_med;

    this.email_med = email_med;

    this.direccion_med = direccion_med;
```

```
    this.telefono = telefono;

    this.activo = activo;

    this.fechanacimiento = fechanacimiento;

    this.fecharegistro = fecharegistro;

    this.esp = esp;

    this.listaReservaciones = listaReservaciones;
}
```

```
public int getId() {

    return id;

}
```

```
public void setId(int id) {

    this.id = id;

}
```

```
public String getNombre_med() {

    return nombre_med;

}
```

```
public void setNombre_med(String nombre_med) {

    this.nombre_med = nombre_med;

}
```

```
public String getApellido_med() {

    return apellido_med;

}
```

```
public void setApellido_med(String apellido_med) {  
    this.apellido_med = apellido_med;  
}
```

```
public String getGenero_med() {  
    return genero_med;  
}
```

```
public void setGenero_med(String genero_med) {  
    this.genero_med = genero_med;  
}
```

```
public String getEmail_med() {  
    return email_med;  
}
```

```
public void setEmail_med(String email_med) {  
    this.email_med = email_med;  
}
```

```
public String getDireccion_med() {  
    return direccion_med;  
}
```

```
public void setDireccion_med(String direccion_med) {  
    this.direccion_med = direccion_med;  
}
```

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

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

```
public Boolean getActivo() {  
    return activo;  
}
```

```
public void setActivo(Boolean activo) {  
    this.activo = activo;  
}
```

```
public Date getFechanacimiento() {  
    return fechanacimiento;  
}
```

```
public void setFechanacimiento(Date fechanacimiento) {  
    this.fechanacimiento = fechanacimiento;  
}
```

```
public Date getFecharegistro() {  
    return fecharegistro;  
}
```

```
public void setFecharregistro(Date fecharregistro) {  
    this.fecharregistro = fecharregistro;  
}
```

```
public Especialidad getEsp() {  
    return esp;  
}
```

```
public void setEsp(Especialidad esp) {  
    this.esp = esp;  
}
```

```
public LinkedList<Reservacion> getListaReservaciones() {  
    return listaReservaciones;  
}
```

```
public void setListaReservaciones(LinkedList<Reservacion> listaReservaciones) {  
    this.listaReservaciones = listaReservaciones;  
}
```

```
@Override
```

```
public String toString() {  
    return "Medico{" + "id=" + id + ", nombre_med=" + nombre_med + ",  
apellido_med=" + apellido_med + ", genero_med=" + genero_med + ", email_med=" +  
email_med + ", direccion_med=" + direccion_med + ", telefono=" + telefono + ",  
activo=" + activo + ", fechanacimiento=" + fechanacimiento + ", fecharregistro=" +  
fecharregistro + ", esp=" + esp + ", listaReservaciones=" + listaReservaciones + '}';  
}*/
```

```
public Medico(Long id, String nombre_med, String apellido_med, String
genero_med, String email_med, String direccion_med, String telefono, Date
fechanacimiento, Date fecharegistro, Especialidad esp) {
```

```
    this.id = id;
```

```
    this.nombre_med = nombre_med;
```

```
    this.apellido_med = apellido_med;
```

```
    this.genero_med = genero_med;
```

```
    this.email_med = email_med;
```

```
    this.direccion_med = direccion_med;
```

```
    this.telefono = telefono;
```

```
    this.fechanacimiento = fechanacimiento;
```

```
    this.fecharegistro = fecharegistro;
```

```
    this.esp = esp;
```

```
}
```

```
public Long getId() {
```

```
    return id;
```

```
}
```

```
public void setId(Long id) {
```

```
    this.id = id;
```

```
}
```

```
public String getNombre_med() {
```

```
    return nombre_med;
```

```
}
```

```
public void setNombre_med(String nombre_med) {  
    this.nombre_med = nombre_med;  
}
```

```
public String getApellido_med() {  
    return apellido_med;  
}
```

```
public void setApellido_med(String apellido_med) {  
    this.apellido_med = apellido_med;  
}
```

```
public String getGenero_med() {  
    return genero_med;  
}
```

```
public void setGenero_med(String genero_med) {  
    this.genero_med = genero_med;  
}
```

```
public String getEmail_med() {  
    return email_med;  
}
```

```
public void setEmail_med(String email_med) {  
    this.email_med = email_med;  
}
```



```
public String getDireccion_med() {  
    return direccion_med;  
}
```

```
public void setDireccion_med(String direccion_med) {  
    this.direccion_med = direccion_med;  
}
```

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

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

```
public Boolean getActivo() {  
    return activo;  
}
```

```
public void setActivo(Boolean activo) {  
    this.activo = activo;  
}
```

```
public Date getFechanacimiento() {  
    return fechanacimiento;  
}
```

```
public void setFechanacimiento(Date fechanacimiento) {  
    this.fechanacimiento = fechanacimiento;  
}
```

```
public java.util.Date getFecharregistro() {  
    return fecharregistro;  
}
```

```
public void setFecharregistro(java.util.Date fecharregistro) {  
    this.fecharregistro = fecharregistro;  
}
```

```
public Especialidad getEsp() {  
    return esp;  
}
```

```
public void setEsp(Especialidad esp) {  
    this.esp = esp;  
}
```

@Override

```
public String toString() {  
    return "Medico{" + "id=" + id + ", nombre_med=" + nombre_med + ",  
apellido_med=" + apellido_med + ", genero_med=" + genero_med + ", email_med=" +  
email_med + ", direccion_med=" + direccion_med + ", telefono=" + telefono + ",  
activo=" + activo + ", fechanacimiento=" + fechanacimiento + ", fecharregistro=" +  
fecharregistro + ", esp=" + esp + '}';  
}
```

```
}
```

- **Paciente**

```
package com.mycompany.proyecto.logica;
```

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

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

```
import javax.persistence.Basic;
```

```
import javax.persistence.Column;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.Table;
```

```
import javax.persistence.Temporal;
```

```
import javax.persistence.TemporalType;
```

```
@Entity
```

```
@Table(name="paciente")
```

```
public class Paciente implements Serializable {
```

```
    @Id
```

```
    //@GeneratedValue(strategy=GenerationType.AUTO)
```

```
    //private int id;
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

@Basic

private String nombrepac;

private String apellidopac;

private String genero;

@Column(unique=true)

private String email;

private String direccion;

private String telefono;

private String enfermedad;

private String alergia;

private String medicamento;

private Boolean activo=true;

@Temporal(TemporalType.DATE)

private Date fechanacimiento;

@Temporal(TemporalType.TIMESTAMP)

private java.util.Date fecharegistro;

public Paciente() {

}

/*public Paciente(int id, String nombrepac, String apellidopac, String genero, String email, String direccion, String telefono, String enfermedad, String alergia, String medicamento, Boolean activo, Date fechanacimiento, Date fecharegistro) {

this.id = id;

this.nombrepac = nombrepac;

this.apellidopac = apellidopac;

this.genero = genero;

this.email = email;

this.direccion = direccion;

```
    this.telefono = telefono;

    this.enfermedad = enfermedad;

    this.alergia = alergia;

    this.medicamento = medicamento;

    this.activo = activo;

    this.fechanacimiento = fechanacimiento;

    this.fecharegistro = fecharegistro;
}
```

```
public int getId() {

    return id;

}
```

```
public void setId(int id) {

    this.id = id;

}
```

```
public String getNombrepac() {

    return nombrepac;

}
```

```
public void setNombrepac(String nombrepac) {

    this.nombrepac = nombrepac;

}
```

```
public String getApellidopac() {

    return apellidopac;

}
```

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

```
public String getGenero() {  
    return genero;  
}
```

```
public void setGenero(String genero) {  
    this.genero = genero;  
}
```

```
public String getEmail() {  
    return email;  
}
```

```
public void setEmail(String email) {  
    this.email = email;  
}
```

```
public String getDireccion() {  
    return direccion;  
}
```

```
public void setDireccion(String direccion) {  
    this.direccion = direccion;  
}
```

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

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

```
public String getEnfermedad() {  
    return enfermedad;  
}
```

```
public void setEnfermedad(String enfermedad) {  
    this.enfermedad = enfermedad;  
}
```

```
public String getAlergia() {  
    return alergia;  
}
```

```
public void setAlergia(String alergia) {  
    this.alergia = alergia;  
}
```

```
public String getMedicamento() {  
    return medicamento;  
}
```

```
public void setMedicamento(String medicamento) {  
    this.medicamento = medicamento;  
}
```

```
public Boolean getActivo() {  
    return activo;  
}
```

```
public void setActivo(Boolean activo) {  
    this.activo = activo;  
}
```

```
public Date getFechanacimiento() {  
    return fechanacimiento;  
}
```

```
public void setFechanacimiento(Date fechanacimiento) {  
    this.fechanacimiento = fechanacimiento;  
}
```

```
public Date getFecharegistro() {  
    return fecharegistro;  
}
```

```
public void setFecharegistro(Date fecharegistro) {  
    this.fecharegistro = fecharegistro;  
}
```


@Override

public String toString() {

**return "Paciente{" + "id=" + id + ", nombrepac=" + nombrepac + ", apellidopac=" +
apellidopac + ", genero=" + genero + ", email=" + email + ", direccion=" + direccion + ",
telefono=" + telefono + ", enfermedad=" + enfermedad + ", alergia=" + alergia + ",
medicamento=" + medicamento + ", activo=" + activo + ", fechanacimiento=" +
fechanacimiento + ", fecharegistro=" + fecharegistro + '}';**

}/

**public Paciente(Long id, String nombrepac, String apellidopac, String genero, String
email, String direccion, String telefono, String enfermedad, String alergia, String
medicamento, Date fechanacimiento, Date fecharegistro) {**

this.id = id;

this.nombrepac = nombrepac;

this.apellidopac = apellidopac;

this.genero = genero;

this.email = email;

this.direccion = direccion;

this.telefono = telefono;

this.enfermedad = enfermedad;

this.alergia = alergia;

this.medicamento = medicamento;

this.fechanacimiento = fechanacimiento;

this.fecharegistro = fecharegistro;

}

public Long getId() {

return id;

}

```
public void setId(Long id) {  
    this.id = id;  
}
```

```
public String getNombrepac() {  
    return nombrepac;  
}
```

```
public void setNombrepac(String nombrepac) {  
    this.nombrepac = nombrepac;  
}
```

```
public String getApellidopac() {  
    return apellidopac;  
}
```

```
public void setApellidopac(String apellidopac) {  
    this.apellidopac = apellidopac;  
}
```

```
public String getGenero() {  
    return genero;  
}
```

```
public void setGenero(String genero) {  
    this.genero = genero;  
}
```

```
public String getEmail() {  
    return email;  
}
```

```
public void setEmail(String email) {  
    this.email = email;  
}
```

```
public String getDireccion() {  
    return direccion;  
}
```

```
public void setDireccion(String direccion) {  
    this.direccion = direccion;  
}
```

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

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

```
public String getEnfermedad() {  
    return enfermedad;  
}
```

```
public void setEnfermedad(String enfermedad) {  
    this.enfermedad = enfermedad;  
}
```

```
public String getAlergia() {  
    return alergia;  
}
```

```
public void setAlergia(String alergia) {  
    this.alergia = alergia;  
}
```

```
public String getMedicamento() {  
    return medicamento;  
}
```

```
public void setMedicamento(String medicamento) {  
    this.medicamento = medicamento;  
}
```

```
public Boolean getActivo() {  
    return activo;  
}
```

```
public void setActivo(Boolean activo) {  
    this.activo = activo;  
}
```

```
public Date getFechanacimiento() {  
    return fechanacimiento;  
}
```

```
public void setFechanacimiento(Date fechanacimiento) {  
    this.fechanacimiento = fechanacimiento;  
}
```

```
public java.util.Date getFecharegistro() {  
    return fecharegistro;  
}
```

```
public void setFecharegistro(java.util.Date fecharegistro) {  
    this.fecharegistro = fecharegistro;  
}
```

@Override

```
public String toString() {  
    return "Paciente{" + "id=" + id + ", nombrepac=" + nombrepac + ", apellidopac=" +  
    apellidopac + ", genero=" + genero + ", email=" + email + ", direccion=" + direccion + ",  
    telefono=" + telefono + ", enfermedad=" + enfermedad + ", alergia=" + alergia + ",  
    medicamento=" + medicamento + ", activo=" + activo + ", fechanacimiento=" +  
    fechanacimiento + ", fecharegistro=" + fecharegistro + '}';  
}
```

```
}
```

- **Especialidades**

```
package com.mycompany.proyecto.logica;
```

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

```
import javax.persistence.Basic;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.Table;
```

```
@Entity
```

```
@Table(name="especialidad")
```

```
public class Especialidad implements Serializable {
```

```
    @Id
```

```
    //@GeneratedValue(strategy = GenerationType.SEQUENCE)
```

```
    //private int id;
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    @Basic
```

```
    private String nombespec;
```

```
    public Especialidad() {
```

```
    }
```

```
/*public Especialidad(int id, String nombespec) {  
  
    this.id = id;  
  
    this.nombespec = nombespec;  
  
}
```

```
public int getId() {  
  
    return id;  
  
}
```

```
public void setId(int id) {  
  
    this.id = id;  
  
}
```

```
public String getNombespec() {  
  
    return nombespec;  
  
}
```

```
public void setNombespec(String nombespec) {  
  
    this.nombespec = nombespec;  
  
}
```

```
@Override
```

```
public String toString() {  
  
    return "Especialidad{" + "id=" + id + ", nombespec=" + nombespec + '}';  
  
}*/
```

```
public Especialidad(Long id, String nombespec) {  
  
    this.id = id;
```

```

        this.nombespec = nombespec;
    }

    public Long getId() {
        return id;
    }

    public void setId(Long id) {
        this.id = id;
    }

    public String getNombespec() {
        return nombespec;
    }

    public void setNombespec(String nombespec) {
        this.nombespec = nombespec;
    }

    @Override
    public String toString() {
        return "Especialidad{" + "id=" + id + ", nombespec=" + nombespec + '}';
    }

}

```

- Reservasiones


```
package com.mycompany.proyecto.logica;
```

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

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

```
import javax.persistence.Basic;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.JoinColumn;
```

```
import javax.persistence.ManyToOne;
```

```
import javax.persistence.Table;
```

```
import javax.persistence.Temporal;
```

```
import javax.persistence.TemporalType;
```

```
@Entity
```

```
@Table(name="reservacion")
```

```
public class Reservacion implements Serializable {
```

```
    @Id
```

```
    //@GeneratedValue(strategy=GenerationType.SEQUENCE)
```

```
    //private int id;
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    @Basic
```

```
    private String titulo;
```

```
    private String nota;
```

```
    private String mensaje;
```

```
    private String sintomas;
```

```
private String enfermedad;

private String medicamento;

private Boolean conectado;

@Temporal(TemporalType.DATE)

private Date fecha;

@Temporal(TemporalType.TIME)

private Date hora;

@Temporal(TemporalType.TIMESTAMP)

private java.util.Date fechacreacion;
```

```
@ManyToOne

@JoinColumn(name="estado_id")

private Estado est;

@ManyToOne

@JoinColumn(name="usuario_id")

private Usuario usu;

@ManyToOne

@JoinColumn(name="medico_id")

private Medico med;

@ManyToOne

@JoinColumn(name="paciente_id")

private Paciente pac;
```

```
public Reservacion() {

}
```

```
/*public Reservacion(int id, String titulo, String nota, String mensaje, String
sintomas, String enfermedad, String medicamento, Boolean conectado, Date
fecha, Date fechacreacion, Estado est, Usuario usu, Medico med, Paciente pac) {
```

```
    this.id = id;

    this.titulo = titulo;

    this.nota = nota;

    this.mensaje = mensaje;

    this.sintomas = sintomas;

    this.enfermedad = enfermedad;

    this.medicamento = medicamento;

    this.conectado = conectado;

    this.fecha = fecha;

    this.hora = hora;

    this.fechacreacion = fechacreacion;

    this.est = est;

    this.usu = usu;

    this.med = med;

    this.pac = pac;
}
```

```
public int getId() {

    return id;

}
```

```
public void setId(int id) {

    this.id = id;

}
```

```
public String getTitulo() {
```

```
        return titulo;
    }
}
```

```
public void setTitulo(String titulo) {
    this.titulo = titulo;
}
}
```

```
public String getNota() {
    return nota;
}
}
```

```
public void setNota(String nota) {
    this.nota = nota;
}
}
```

```
public String getMensaje() {
    return mensaje;
}
}
```

```
public void setMensaje(String mensaje) {
    this.mensaje = mensaje;
}
}
```

```
public String getSintomas() {
    return sintomas;
}
}
```

```
public void setSintomas(String sintomas) {
```

```
    this.sintomas = sintomas;  
}
```

```
public String getEnfermedad() {  
    return enfermedad;  
}
```

```
public void setEnfermedad(String enfermedad) {  
    this.enfermedad = enfermedad;  
}
```

```
public String getMedicamento() {  
    return medicamento;  
}
```

```
public void setMedicamento(String medicamento) {  
    this.medicamento = medicamento;  
}
```

```
public Boolean getConectado() {  
    return conectado;  
}
```

```
public void setConectado(Boolean conectado) {  
    this.conectado = conectado;  
}
```

```
public Date getFecha() {
```

```
        return fecha;
    }
}
```

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

```
public Date getHora() {
    return hora;
}
}
```

```
public void setHora(Date hora) {
    this.hora = hora;
}
}
```

```
public Date getFechacreacion() {
    return fechacreacion;
}
}
```

```
public void setFechacreacion(Date fechacreacion) {
    this.fechacreacion = fechacreacion;
}
}
```

```
public Estado getEst() {
    return est;
}
}
```

```
public void setEst(Estado est) {
```

```
        this.est = est;
    }
}
```

```
public Usuario getUsu() {
    return usu;
}
```

```
public void setUsu(Usuario usu) {
    this.usu = usu;
}
```

```
public Medico getMed() {
    return med;
}
```

```
public void setMed(Medico med) {
    this.med = med;
}
```

```
public Paciente getPac() {
    return pac;
}
```

```
public void setPac(Paciente pac) {
    this.pac = pac;
}
```

```
@Override
```

```

public String toString() {

    return "Reservacion{" + "id=" + id + ", titulo=" + titulo + ", nota=" + nota + ",
mensaje=" + mensaje + ", sintomas=" + sintomas + ", enfermedad=" + enfermedad + ",
medicamento=" + medicamento + ", conectado=" + conectado + ", fecha=" + fecha + ",
fechacreacion=" + fechacreacion + ", est=" + est + ", usu=" + usu + ", med=" + med + ",
pac=" + pac + '}';

}*/

```

```

public Reservacion(Long id, String titulo, String nota, String mensaje, String
sintomas, String enfermedad, String medicamento, Boolean conectado, Date fecha,
Date hora, Date fechacreacion, Estado est, Usuario usu, Medico med, Paciente pac) {

```

```

    this.id = id;

    this.titulo = titulo;

    this.nota = nota;

    this.mensaje = mensaje;

    this.sintomas = sintomas;

    this.enfermedad = enfermedad;

    this.medicamento = medicamento;

    this.conectado = conectado;

    this.fecha = fecha;

    this.hora = hora;

    this.fechacreacion = fechacreacion;

    this.est = est;

    this.usu = usu;

    this.med = med;

    this.pac = pac;

}

```

```

public Long getId() {

    return id;

```



```
}
```

```
public void setId(Long id) {
```

```
    this.id = id;
```

```
}
```

```
public String getTitulo() {
```

```
    return titulo;
```

```
}
```

```
public void setTitulo(String titulo) {
```

```
    this.titulo = titulo;
```

```
}
```

```
public String getNota() {
```

```
    return nota;
```

```
}
```

```
public void setNota(String nota) {
```

```
    this.nota = nota;
```

```
}
```

```
public String getMensaje() {
```

```
    return mensaje;
```

```
}
```

```
public void setMensaje(String mensaje) {
```

```
    this.mensaje = mensaje;
```

```
}
```

```
public String getSintomas() {  
    return sintomas;  
}
```

```
public void setSintomas(String sintomas) {  
    this.sintomas = sintomas;  
}
```

```
public String getEnfermedad() {  
    return enfermedad;  
}
```

```
public void setEnfermedad(String enfermedad) {  
    this.enfermedad = enfermedad;  
}
```

```
public String getMedicamento() {  
    return medicamento;  
}
```

```
public void setMedicamento(String medicamento) {  
    this.medicamento = medicamento;  
}
```

```
public Boolean getConectado() {  
    return conectado;
```

```
}
```

```
public void setConectado(Boolean conectado) {
```

```
    this.conectado = conectado;
```

```
}
```

```
public Date getFecha() {
```

```
    return fecha;
```

```
}
```

```
public void setFecha(Date fecha) {
```

```
    this.fecha = fecha;
```

```
}
```

```
public Date getHora() {
```

```
    return hora;
```

```
}
```

```
public void setHora(Date hora) {
```

```
    this.hora = hora;
```

```
}
```

```
public java.util.Date getFechacreacion() {
```

```
    return fechacreacion;
```

```
}
```

```
public void setFechacreacion(java.util.Date fechacreacion) {
```

```
    this.fechacreacion = fechacreacion;
```

```
}
```

```
public Estado getEst() {
```

```
    return est;
```

```
}
```

```
public void setEst(Estado est) {
```

```
    this.est = est;
```

```
}
```

```
public Usuario getUsu() {
```

```
    return usu;
```

```
}
```

```
public void setUsu(Usuario usu) {
```

```
    this.usu = usu;
```

```
}
```

```
public Medico getMed() {
```

```
    return med;
```

```
}
```

```
public void setMed(Medico med) {
```

```
    this.med = med;
```

```
}
```

```
public Paciente getPac() {
```

```
    return pac;
```

```
}
```

```
public void setPac(Paciente pac) {
```

```
    this.pac = pac;
```

```
}
```

```
@Override
```

```
public String toString() {
```

```
    return "Reservacion{" + "id=" + id + ", titulo=" + titulo + ", nota=" + nota + ",  
mensaje=" + mensaje + ", sintomas=" + sintomas + ", enfermedad=" + enfermedad + ",  
medicamento=" + medicamento + ", conectado=" + conectado + ", fecha=" + fecha + ",  
hora=" + hora + ", fechacreacion=" + fechacreacion + ", est=" + est + ", usu=" + usu + ",  
med=" + med + ", pac=" + pac + '}';
```

```
}
```

```
}
```

- Estado

```
package com.mycompany.proyecto.logica;
```

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

```
import java.util.List;
```

```
import javax.persistence.Basic;
```

```
import javax.persistence.Column;
```

```
import javax.persistence.Entity;
```

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

```
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
```

```
import javax.persistence.Id;

import javax.persistence.Query;

import javax.persistence.Table;


@Entity

@Table(name="estado")

public class Estado implements Serializable {

    @Id

    // @GeneratedValue(strategy= GenerationType.AUTO)

    // private int id;

    @GeneratedValue(strategy = GenerationType.IDENTITY)

    @Column(name="id")

    private Long id;

    @Basic

    @Column(name="tipo")

    private String tipo;


    public Estado() {

    }


    /*public Estado(int id, String estado) {

        this.id = id;

        this.estado = estado;

    }


    public int getId() {

        return id;

    }

}
```

```
public void setId(int id) {  
    this.id = id;  
}
```

```
public String getEstado() {  
    return estado;  
}
```

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

```
@Override
```

```
public String toString() {  
    return "Estatus{" + "id=" + id + ", estado=" + estado + '}';  
}*/
```

```
public Estado(String tipo) {  
    this.tipo = tipo;  
}
```

```
public Long getId() {  
    return id;  
}
```

```
public void setId(Long id) {  
    this.id = id;
```

```
}
```

```
public String getTipo() {
```

```
    return tipo;
```

```
}
```

```
public void setTipo(String tipo) {
```

```
    this.tipo = tipo;
```

```
}
```

```
public static List<Estado> obtenerTiposDeEstados(EntityManager entityManager) {
```

```
    Query query = entityManager.createQuery("SELECT e FROM Estado e");
```

```
    return query.getResultList();
```

```
}
```

```
@Override
```

```
public String toString() {
```

```
    return "Estado{" +
```

```
        "id=" + id +
```

```
        ", tipo=" + tipo + '\n' +
```

```
        '}';
```

```
}
```

```
}
```

7. POM.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```



```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<groupId>com.mycompany</groupId>
```

```
<artifactId>proyecto</artifactId>
```

```
<version>1.0-SNAPSHOT</version>
```

```
<packaging>jar</packaging>
```

```
<dependencies>
```

```
<dependency>
```

```
<groupId>com.mysql</groupId>
```

```
<artifactId>mysql-connector-j</artifactId>
```

```
<version>8.0.33</version>
```

```
</dependency>
```

```
<dependency>
```

```
<groupId>org.eclipse.persistence</groupId>
```

```
<artifactId>org.eclipse.persistence.core</artifactId>
```

```
<version>2.7.10</version>
```

```
</dependency>
```

```
<dependency>
```

```
<groupId>org.eclipse.persistence</groupId>
```

```
<artifactId>org.eclipse.persistence.asm</artifactId>
```

```
<version>9.2.0</version>
```

```
</dependency>
```

```
<dependency>
```

```
<groupId>org.eclipse.persistence</groupId>
```

```
<artifactId>org.eclipse.persistence.antlr</artifactId>
```

```
<version>2.7.10</version>
```

</dependency>

<dependency>

<groupId>org.eclipse.persistence</groupId>

<artifactId>org.eclipse.persistence.jpa</artifactId>

<version>2.7.10</version>

</dependency>

<dependency>

<groupId>org.eclipse.persistence</groupId>

<artifactId>org.eclipse.persistence.jpa.jpql</artifactId>

<version>2.7.10</version>

</dependency>

<dependency>

<groupId>org.eclipse.persistence</groupId>

<artifactId>org.eclipse.persistence.moxy</artifactId>

<version>2.7.10</version>

</dependency>

<dependency>

<groupId>org.eclipse.persistence</groupId>

<artifactId>jakarta.persistence</artifactId>

<version>2.2.3</version>

</dependency>

<dependency>

<groupId>org.eclipse.persistence</groupId>

<artifactId>org.eclipse.persistence.jpa.modelgen.processor</artifactId>

<version>2.7.10</version>

<scope>provided</scope>

</dependency>

<dependency>

```
<groupId>org.netbeans.external</groupId>  
  
<artifactId>AbsoluteLayout</artifactId>  
  
<version>RELEASE190</version>  
  
</dependency>  
</dependencies>
```

```
<properties>  
  
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
  
<maven.compiler.source>17</maven.compiler.source>  
  
<maven.compiler.target>17</maven.compiler.target>  
  
<exec.mainClass>com.mycompany.proyecto.Proyecto</exec.mainClass>  
  
</properties>  
</project>
```