

PROYECT

O

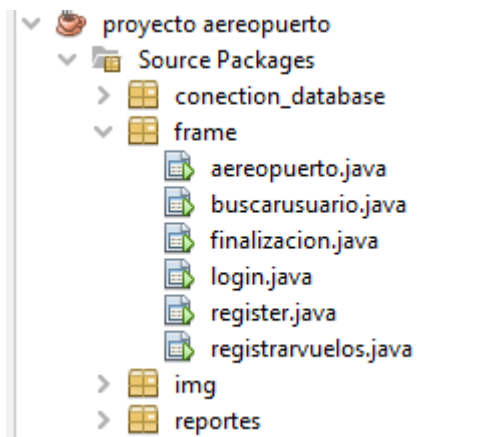
AEREOPUE

RTO

VERSION

(2.0)

JFRAMES USADOS EN EL PROYECTO



ID USADO :NETBEANS 15

Busqueda de usuarios registrados

Ingrese su usuario a buscar

Buscar

PROGRAMAMOS EL BOTON BUSCAR EN LA DATABASE

```
private void enviarActionPerformed(java.awt.event.ActionEvent evt) {  
    Connection conexion=null;  
    try {  
        conexion=iniciarconexion();  
        String consulta=("select*from usuarios where usuario=? ");  
        PreparedStatement statement=conexion.prepareStatement(consulta);  
        statement.setString(1, usuario.getText());  
        rs=statement.executeQuery();  
        if(rs.next()){  
            usuario.setText(rs.getString("usuario"));  
            JOptionPane.showMessageDialog(null, "usuario encontrado");  
        }else{
```

```
JOptionPane.showMessageDialog(null, "usuario no encontrado (puede registrarse)");  
}  
  
} catch (Exception e) {  
    System.err.println("error "+e);  
}
```

REGISTER

Name:

Las Name:

Number

DNI

Mail

User:

Password: [Generar contraseña](#)

PROGRAMAMOS EL BOTON (GENERAR CONTRASEÑAS)

```

private void generarcontraseña(){
    String contra="";
    contra=UUID.randomUUID().toString().toUpperCase().substring(0,10);
    contraseña.setText(contra);
}

```

CONFIGURAMOS EL BOTON SEND PARA HACER UN REGISTRO A LA BASE DE DATOS

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    Connection conex=null;
    try {
        conex=iniciarConnection();

        String consulta=("insert into
usuarios(nombre,apellido,numero,correo,dni,usuario,contraseña) values(?,?,?,?,?,?,?)");

        PreparedStatement statement= conex.prepareStatement(consulta);

        statement.setString(1, nombre.getText());
        statement.setString(2, apellido.getText());
        statement.setString(3, number.getText());
        statement.setString(4, gmail.getText());
        statement.setString(5, String.valueOf(dni.getText()));
        statement.setString(6, usuario.getText());
        statement.setString(7, contraseña.getText());

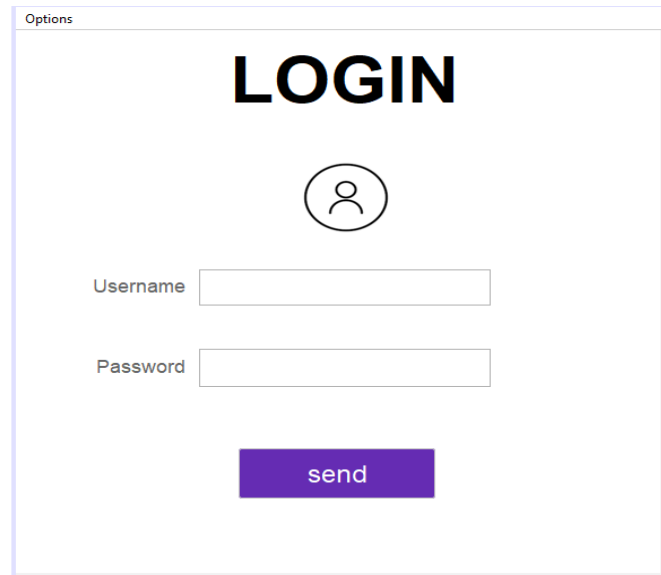
        int resultado=statement.executeUpdate();
        if(resultado>0){
            JOptionPane.showMessageDialog(null, "usuario registrado");
        }else{
            JOptionPane.showMessageDialog(null, "ERROR ");
        }
    }
}

```

```
    } catch (Exception e) {  
        System.err.println("error"+e);  
    }  
}
```

HACEMOS VISIBLE LA OTRA VENTANA

```
private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {  
    login log=new login();  
    log.setVisible(true);  
    this.dispose();  
}
```

INIALIZACION DE CODIGO PARA LOS FRAME

```
public class login extends javax.swing.JFrame {
```

```
    ResultSet rs; :OBJETO RESULTSET
```

CONEXION A LA BASE DE DATOS “REGISTROUSER”

```
    public static Connection iniciarConnection(){
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection co= DriverManager.getConnection("jdbc:mysql://localhost:3306/registrouser", "root",
"1234");
            Statement stm = co.createStatement();
            return co;
        } catch (ClassNotFoundException e) {
            System.out.println("Clase no encontrada: "+e);
        } catch (SQLException e) {
            System.out.println("Error de conexion: "+e);
        }
        return null;
    }
}
```

HACEMOS VISIBLE EL JFrame REGI

```
    private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {
```



```

register regi=new register();
regi.setVisible(true);
this.dispose();
}

```

PROGRAMAMOS EL BOTON DE ENVIAR

```

private void sendActionPerformed(java.awt.event.ActionEvent evt) {
    Connection conexion=null;
    try {
        conexion=iniciarConnection();
        String consulta=("select*from usuarios where usuario=? and contraseña=? ");
        PreparedStatement statement= conexion.prepareStatement(consulta);
        statement.setString(1, username.getText());
        String contraseña="";
        for(int i =0;i<contra.getPassword().length;i++){
            contraseña+=contra.getPassword()[i];
        }
        statement.setString(2, contraseña);
        rs=statement.executeQuery();
        if(rs.next()){
            username.setText(rs.getString("usuario"));
            contra.setText(rs.getString("contraseña"));
            aereopuerto aero =new aereopuerto();
            aero.setVisible(true);
            this.dispose();
        }else{
            JOptionPane.showMessageDialog(null,"Error compruebe su contraseña o usuario");
        }
    } catch (Exception e) {
        System.err.println("error"+e);
    }
}

```

PROGRAMAMOS EL BOTON BUSQUEDA

```
private void busquedaActionPerformed(java.awt.event.ActionEvent evt) {  
    buscarusuario user=new buscarusuario();  
    user.setVisible(true);  
    this.dispose();  
}
```

Aereopuertos:Disponibles

Nombre	Ubicacion
Iuna	Piura
Rios	Cajamarca
Aero Peru	Lima
Peru S.A.C	Lima
Iuna	Piura
Rios	Cajamarca
Grisfed	Arequipa
Rios	Cajamarca
Aero Peru	Lima

NOMBRE	<input type="text" value="Peru S.A.C"/>	ASIENTOS DISPONIBLES	<input type="text" value="55"/>
VUELO	<input type="text" value="G8LO9"/>	COMPAÑIAS	<input type="text" value="aphia"/>
DESTINO	<input type="text" value="Alemania"/>	Costo de vuelo	<input type="text" value="450"/>

BOTON REGISTRA VUELOS

```
private void registrarvueloActionPerformed(java.awt.event.ActionEvent evt) {  
    registrarvuelos vuelos=new registrarvuelos();  
    vuelos.setdato(NOMBRE.getText());  
    vuelos.setdate(DESTINO.getText());  
    vuelos.obtenerprecio(precio.getText());  
    vuelos.setVisible(true);  
    this.dispose();  
}
```

```
}
```

BOTON CARGAR AEREOPUERTO

```
private void cargar_aereopuertosActionPerformed(java.awt.event.ActionEvent evt) {  
DefaultTableModel modelo=new DefaultTableModel();  
tablita.setModel(modelo);  
  
Connection conexion=null;  
ResultSet rs=null;  
  
try {  
    conexion=iniciarConnection();  
    String consulta=("select nombre,ubi from aereopuertos");  
    PreparedStatement statement=conexion.prepareStatement(consulta);  
    rs=statement.executeQuery();  
    modelo.addColumn("Nombre");//esdtblecemos el texto a la clumna  
    modelo.addColumn("Ubicacion");  
    while(rs.next()){//agregar datos a la tabla  
        Object fila[]=new Object[2];//creamos un arreglo fila  
        for(int i =0;i<2;i++){  
            fila[i]=rs.getObject(i+1);  
        }  
        modelo.addRow(fila);  
    }  
  
} catch (Exception e) {  
    System.err.println("error "+e);  
}  
}
```

CONFIGURAMOS LA SELECCION DE LA TABLA

```
private void tablitaMouseClicked(java.awt.event.MouseEvent evt) {  
    PreparedStatement ps=null;  
    ResultSet rs=null;  
    Connection conexion=null;  
    try {  
        conexion=iniciarConnection();  
        int seleccion=tablita.getSelectedRow();  
        String eleccion=tablita.getValueAt(seleccion, 0).toString();//obtenemos la fila  
seleccionada  
        ps=conexion.prepareStatement("select  
nombre,numvue,destino,asientosdisponibles,compañia,aeropuertocol,precio from  
aeropuertos where nombre=?");  
        ps.setString(1, eleccion);//y hacemos concordar con el nombre la fila seleccionada  
        rs=ps.executeQuery();  
  
        while(rs.next()){  
            NOMBRE.setText(rs.getString("nombre"));  
            NUMERO_VUELOS.setText(rs.getString("numvue"));  
            DESTINO.setText(rs.getString("destino"));  
            ASIENTOS_DISPONIBLES.setText(rs.getString("asientosdisponibles"));  
            COMPAÑIAS1.setText(rs.getString("compañia"));  
            precio.setText(rs.getString("precio"));  
        }  
  
    } catch (Exception e) {  
        System.err.println("error "+e);  
    }  
}
```

REGISTRAR VUELOS

Options

Registro para vuelo

26/02/2023

Aereopuerto selecionado Peru S.A.C

Con destino hacia: Alemania

Precio del vuelo: 450

Nombre

Apellido

DNI

Correo(Gmail)

Pais

Nacionalidad

send

OBTENCION DE DATOS DEL FRAME ANTERIOR

```
public void setdato(String dato){
    this.dato=dato;
    aereopuerto_seleccionado.setText(dato);
}

public void setdate(String date){
    this.date=date;
    destino_seleccionado1.setText(date);
}

public void obtenerprecio(String precio){
    this.precio=precio;
    totalç.setText(precio);
}

}
```

```

public registrarvuelos() {
    initComponents();
    setLocationRelativeTo(null);
    fecha.setText(fecha());
}

```

CREAMOS UN ARCHIVO QUE SE REGISTRARA LOS DATOS MUY APARTE ES LA BASE DE DATOS

```

public void creararchivo(){
    File archivo=new File("archivo.txt");
    try {
        if(archivo.createNewFile()){
            System.out.println("se creo coreactaente");
        }
    } catch (Exception e) {
        System.out.println("error "+e);
    }
}

```

ESCRIBIMOS EL ARCHIVO CON LOS DATOS

```

public void escribirarchivo(){
    try {
        FileWriter escribir =new FileWriter("archivo.txt",true);
        escribir.write("REGISTRO DE USUARIOS"+"\\r\\nNombre:"+nombre.getText()+
            "\\r\\n"+"Apellido:"+apellido.getText()+
            "\\r\\n"+"DNI:"+dni.getText()+
            "\\r\\n"+"Correo:"+correo.getText()+
            "\\r\\n"+"Pais:"+pais.getText()+
            "\\r\\n"+"nacionalidad.getText()+
            "\\r\\n"+"Aeropuerto seleccionado:"+aeropuerto_seleccionado.getText()+
            "\\r\\n"+"Destino del vuelo:"+destino_seleccionado1.getText()

```

```

        +"\r\n"+"Total a pagar:"+totalç.getText()+"\r\n\r\n");
    escribir.close();
} catch (Exception e) {
    System.err.println("error"+e);
}

```

HACEMOS QUE LA SIGUIENTE VENTANA SEA VISIBLE

```

private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {
    aereopuerto aero=new aereopuerto();
    aero.setVisible(true);
    this.dispose();
}

```

PROGRAMAMOS EL BOTON ENVIAR

```

private void enviarActionPerformed(java.awt.event.ActionEvent evt) {
    Connection conexion=null;
    try {
        conexion=iniciarConnection();

        String consulta =("insert into
compravuelos(nombre,apellido,dni,correo,pais,nacionalidad,precio) values
(?,?,?,?,?,?,?)");

        PreparedStatement statement=conexion.prepareStatement(consulta);
        statement.setString(1, nombre.getText());
        statement.setString(2, apellido.getText());
        statement.setString(3, dni.getText());
        statement.setString(4, correo.getText());
        statement.setString(5, pais.getText());
        statement.setString(6, nacionalidad.getText());
        statement.setString(7, totalç.getText());
    }
}

```

```
int resultado=statement.executeUpdate();
if(resultado>0){
    JOptionPane.showMessageDialog(null, "registro de vuelo exitoso");
    finalizacion fin=new finalizacion();
    fin.setVisible(true);
    fin.obtenernombre(nombre.getText(), apellido.getText(), dni.getText(),
correo.getText(), pais.getText(), nacionalidad.getText(), destino_seleccionado1.getText(),
aeropuerto_seleccionado.getText() );
    escribirarchivo();
    fin.obtenertotal(totalç.getText());
    this.dispose();
    }else{
        JOptionPane.showMessageDialog(null, "ERROR ");
    }
} catch (Exception e) {
    System.err.println("error"+e);
}
}
```


COMPROVANTE DE COMPRA

Nombre

Apellido

DNI

Correo(Gmail)

País

Nacionalidad

Destino

Aeropuerto

TOTAL A PAGAR

Generar pdf

CONIGURAMOS LOS ENVIOS DE DATOS DEL FRAME ANTERIOR PARA LOS JLABEL DE ESTE FRAME

```
String name,ape,documento,corr,pa,nacio,desti,aereopuert,tota;
```

```
public finalizacion() {
```

```
    initComponents();
```

```
    setLocationRelativeTo(null);
```

```
}
```

```
public void obtenertotal(String tota){
```

```
    this.tota=tota;
```

```
    total.setText(tota);
```

```
}
```

```
public void obtenernombre(String name, String ape, String documento, String corr, String pa, String nacio, String desti, String aereopuert) {
```

```
    this.name = name;
```

```
    nombre.setText(name);
```

```
    this.ape = ape;
```

```
    apellido.setText(ape);
```

```

this.documento = documento;
dni.setText(documento);
this.corr = corr;
correo.setText(corr);
this.pa = pa;
pais.setText(pa);
this.nacio = nacio;
nacionalidad.setText(nacio);
this.desti = desti;
desnito.setText(desti);
this.aereopuert = aereopuert;
aereopuerto.setText(aereopuert);

```

GENERAMOS PDF DE TODO LOS REGISTROS (VERSION OPCIONAL)

```

private void generar_pdfActionPerformed(java.awt.event.ActionEvent evt) {
    Connection conexion=null;
    try {
        conexion=iniciarConnection();
        JasperReport reporte=null;//creamos un objeto jasperreposter
        String ruta="src\\reportes\\reportes.jasper";//busacmos la ruta del reporte
        reporte=(JasperReport) JRLoader.loadObjectFromFile(ruta);//agregamos la ruta al erporte y
        le haces un casteo
        JasperPrint imprimirreporte=JasperFillManager.fillReport(reporte, null,
        conexion);//impirmimos el reporte
        //pasamos el reporte que imprimimos ,x ahora null xq queremos
        mostrar todos,conexion a la database

        JasperViewer vistareporte=new JasperViewer(imprimirreporte,false);//hacemos visible al
        //reporte (parametros pasamos el reporte que queremos ver ,false le pasamos siempre false
        ya
        //que si es true al momneto de cerrar el reporte see cerrara el ptoograma con false no

        vistareporte.setDefaultCloseOperation(DISPOSE_ON_CLOSE);//cerrar y no trabar

```

```
        vistareporte.setVisible(true);//hacemos visible al reporte  
    } catch (Exception e) {  
    }
```

**LOS PROYECTOS SERAN GUARDADOS EN UN USB DE NOMBRE PROYEC EN LA
CARPETA PPROYECTOS JAVA**