FORMATO DE INFORME DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA ESTUDIANTES

CARRERA:Ing Computacion	**ASIGNATURA**: Programacion Aplicada
NRO. PRÁCTICA:	**TÍTULO PRÁCTICA**: Prueba Enunciado JPA

OBJETIVO ALCANZADO:
Reforzar los conocimientos adquiridos en clase sobre JPA en un contexto real.

ACTIVIDADES

Objetivo:

- Consolidar los conocimientos adquiridos en clase sobre JPA.

Enunciado:

Realizar un sistema implementando todos los conceptos vistos en clases para gestionar la hipoteca de las casas con las siguientes características:

- Las personas compran casas y se convierten en propietarios.
- Para pagarlas es habitual que el propietario formalice un préstamo hipotecario con una entidad bancaria.e
- El banco toma la casa en forma de aval en caso de impago de las mensualidades.
- En el caso de que el capital fiado supera el valor de tasación de la casa y el sueldo del propietario no es suficiente, el banco suele exigir la presencia de un avalista (garante).
- Para formalizar la hipoteca se necesitan los datos personales del propietario, además de su cédula , dirección de la casa, su dirección, nombres, apellidos y fecha de nacimiento y del garante de ser necesario.
- El capital de la hipoteca se ajusta teniendo en cuenta el valor de tasación de la casa y los datos de dirección.
- Toda hipoteca se formaliza detallando el capital, el interés (8,99 - 16,99%) y la duración (fecha de inicio y fecha de fin).
- A partir de estos datos se calcula el importe de cada mensualidad para el total del tiempo que pide el préstamo .
- No es necesario guardar los datos del banco pero si un sistema de autentificación.
- Generar los datos con el sistema de amortización Alemán [1].

Ejemplo simulador de crédito para guiá: https://www.pichincha.com/portal/simuladores/simulador-de-

creditos

Se calificará de la siguiente forma:

- JPA: 40%
- Excepciones: 10%
- MVC: 20%
- Diagrama de clases: 10%
- Usabilidad - Vista: 15%
- Programación genérica, Java 8, reflexión: 15%

Bibliografía

- [1] https://www.produbanco.com.ec/banca-minorista/cr%C3%A9ditos/hipotecario/simulador-de-cr%C3%A9dito-hipotecario/

Entrega: Subir al Git el documento en formato PDF de los resultados y código hasta las ***23:55*** del domingo ***31 de Enero del 2021***.

1. Para el desarrollo del proyecto se procedió primeo ala investigación sobre el sistema de prestamos dentro de un banco en los enlaces facilitados por el docente, de esta manera se llego a la premisa de la creación de un proyecto donde se evidencia en JPA y para ellos se muestra a continuación de una manera mas detallada.
   Creación del proyecto dentro de la carpeta PruebasSegundoInterciclo_PA llamado Prueba_BD

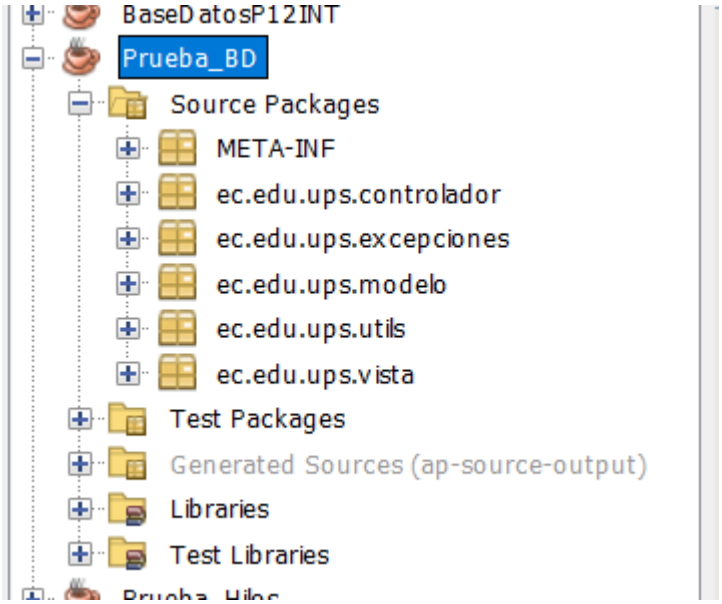
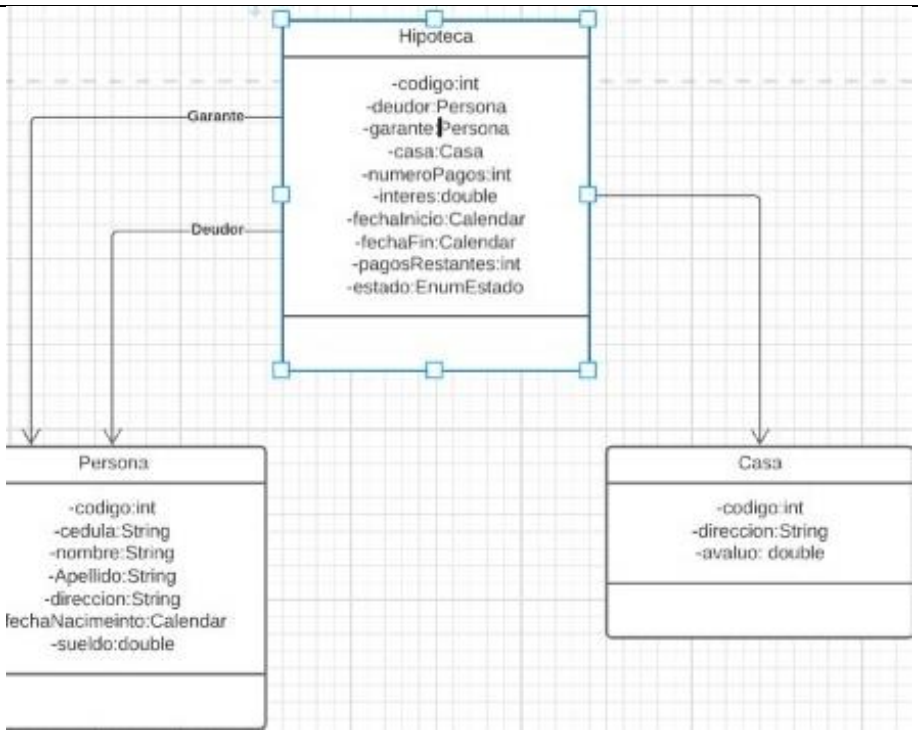
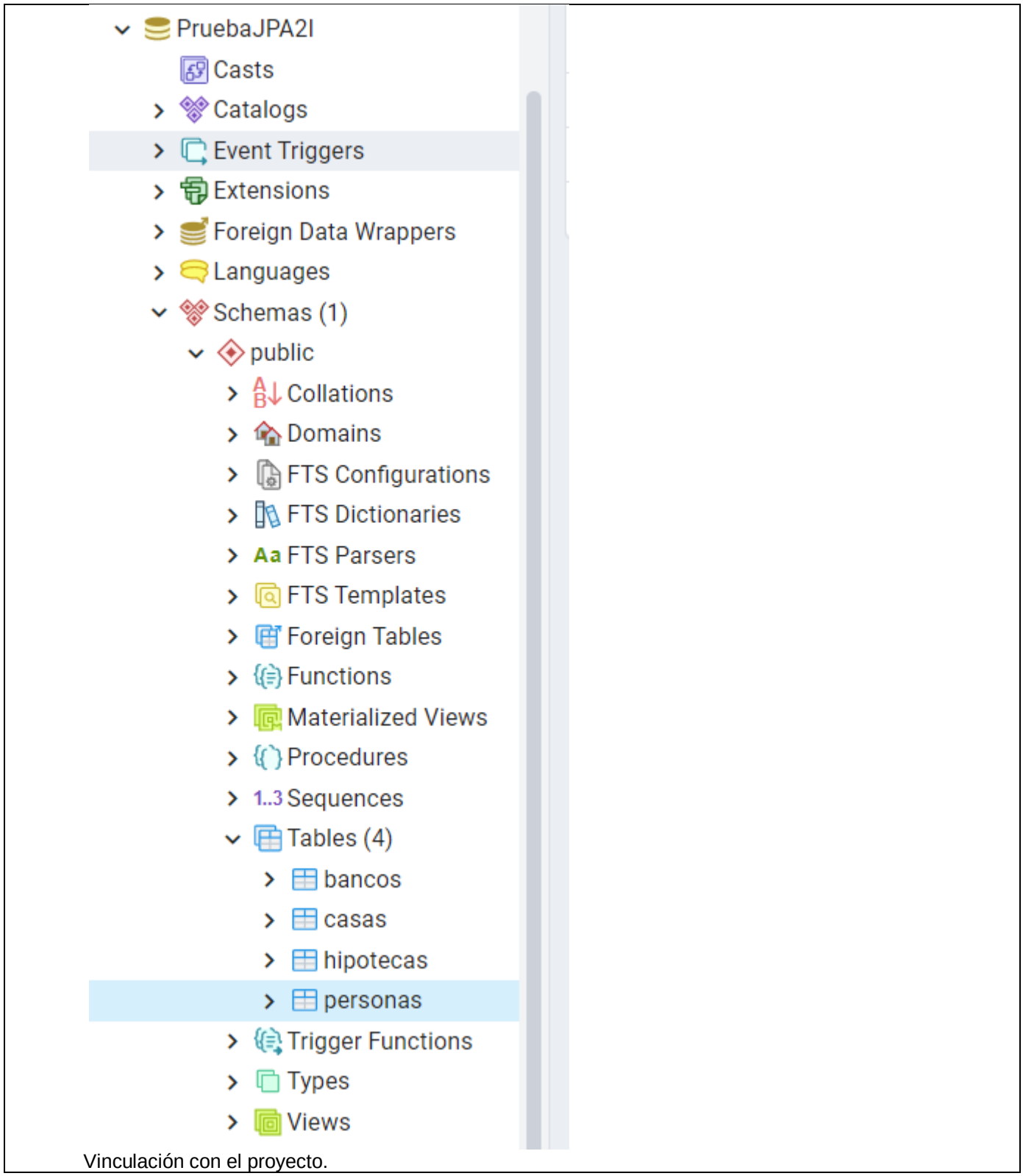


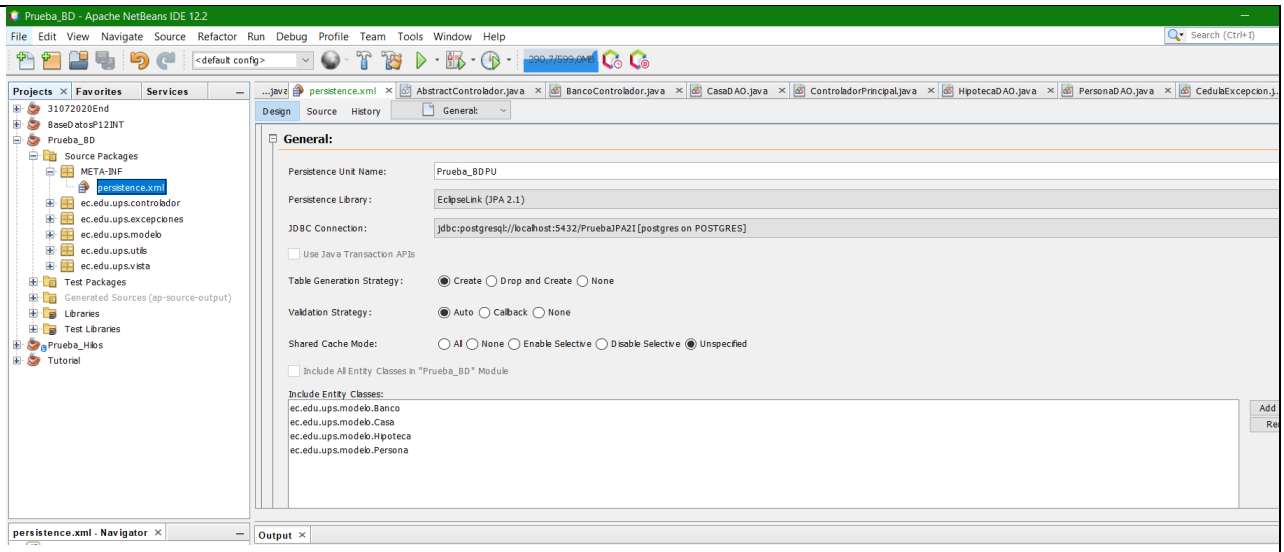
   Diagrama UML.



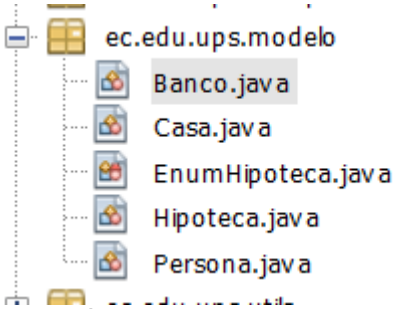
Creación de un Data base llamado Prueba JPA2I



Vinculación con el proyecto.



Codificación de las clases dentro del paquete modelo se procedió a crear clases detalladas a continuación.



Creación de la clase Banco donde esta comentado en algunas líneas de código la funcionalidad y el propósito de si mismo.

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.modelo;

import java.io.Serializable;
import java.util.List;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.OneToMany;
import javax.persistence.Table;



/**
 * entidy es la base de datos con la que se  vincula el proyecto
 * @author ASUS
 */
@Entity
@Table(name = "bancos")
@NamedQueries({
    @NamedQuery(name = "Banco.findAll", query = "SELECT b FROM Banco b")

})
public class Banco implements Serializable{
    /**
     * crea las columnaas en la base de datos
     */
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "codigo")
    private int codigo;
    @Column(name="nombre")
    private String nombre;


    @OneToMany(mappedBy = "banco")
```

```
    private List<Hipoteca> hipoteca;

    public Banco() {

    }

    public int getCodigo() {
        return codigo;
    }

    public void setCodigo(int codigo) {
        this.codigo = codigo;
    }

    public String getNombre() {
        return nombre;
    }

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



    @Override
    public int hashCode() {
        int hash = 5;
        hash = 37 * hash + this.codigo;
        return hash;
    }
```

```
    @Override
    public boolean equals(Object obj) {
        if (this == obj) {
            return true;
        }
        if (obj == null) {
            return false;
        }
        if (getClass() != obj.getClass()) {
            return false;
        }
        final Banco other = (Banco) obj;
        if (this.codigo != other.codigo) {
            return false;
        }
        return true;
    }


}
```

Creación de la clase casa detallada a continuacion.

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.modelo;


import java.io.Serializable;
import javax.persistence.Column;
import javax.persistence.Entity;
```

```
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;


/**
 *
 * @author ASUS
 * damed queries nos sirve para buscar las tablas
 */
@Entity
@Table(name = "casas")
@NamedQueries({
    @NamedQuery(name = "Casa.findAll", query = "SELECT c FROM Casa c"),
    @NamedQuery(name = "Casa.findByCodigo", query = "SELECT c FROM Casa c WHERE c.codigo =
:codigo"),
})
public class Casa implements Serializable {


    @Id
    //se genera un codigo por defecto
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    //creamos la columna
    @Column(name = "codigo")
    private int codigo;
    @Column(name = "Direccion")
    private String direccion;
    @Column(name = "avaluo")
```

```
    private double avaluo;
    @Column(name = "estado")
    private boolean estado;
    public Casa() {

    }

    public Casa(String direccion, double avaluo,boolean estado) {
        this.direccion = direccion;
        this.avaluo = avaluo;
        this.estado = estado;
    }

    public int getCodigo() {
        return codigo;
    }

    public void setCodigo(int codigo) {
        this.codigo = codigo;
    }

    public String getDireccion() {
        return direccion;
    }

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

    public double getAvaluo() {
```

```
    return avaluo;
}

public void setAvaluo(double avaluo) {
    this.avaluo = avaluo;
}

public boolean getEstado() {
    return estado;
}

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

@Override
public String toString() {
    return " Direccion=" + direccion + "\n Avaluo=" + avaluo + '$';
}

@Override
public int hashCode() {
    int hash = 3;
    hash = 97 * hash + this.codigo;
    return hash;
}

@Override
public boolean equals(Object obj) {
    if (this == obj) {
```

```
            return true;
        }
        if (obj == null) {
            return false;
        }
        if (getClass() != obj.getClass()) {
            return false;
        }
        final Casa other = (Casa) obj;
        if (this.codigo != other.codigo) {
            return false;
        }
        return true;
    }


}
```

creación de la clase hipoteca detallada a continuación con alguno de sus atributos y con sus respectivos métodos get and set asi como el tuString

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.modelo;


import java.io.Serializable;
import java.util.Calendar;
import java.util.Date;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.EnumType;
```

```
import javax.persistence.Enumerated;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.OneToOne;
import javax.persistence.Table;
import javax.persistence.Temporal;


/**
 *
 * @author ASUS
 */
@Entity
@Table(name ="hipotecas")
@NamedQueries({
    //query son como buscadors
    @NamedQuery(name = "Hipoteca.findAll", query = "SELECT h FROM Hipoteca h"),
    @NamedQuery(name = "Hipoteca.findByCodigo", query = "SELECT h FROM Hipoteca h WHERE h.codigo = :codigo"),
})
public class Hipoteca  implements  Serializable{


    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "codigo")
    private int codigo;
```

```
@Column(name = "monto")
private double monto;
@Column(name = "interes")
private double interes;
@Column(name = "fecha_Inicio")
// es para poder ocupar una varibale de tipo DATE
@Temporal(javax.persistence.TemporalType.DATE)
private Calendar fechaInicio;
@Column(name = "fecha_Fin")
@Temporal(javax.persistence.TemporalType.DATE)
private Calendar fechaFin;
@Column(name = "pago_mensualidad")
private double pagoMensualidad;
@OneToOne
//Join sirve para relacionar un objeto con otro tecnicamente llamada una clave foraanea
@JoinColumn(name = "casa_id", nullable = false, referencedColumnName = "codigo")
private Casa casa;
@Column(name = "numero_pagos")
private int numeroPagos;
@Column(name = "pagos_Faltantes")
private int pagosFaltantes;
@OneToOne
@JoinColumn(name = "deudor", nullable = false,referencedColumnName = "cedula" )
private Persona deudor;
@OneToOne
@JoinColumn(name = "garante",referencedColumnName = "cedula")
private Persona garante;
@Column(name="Estado")
//SON PARA LA CLASE ENUMhIPOTECA
@Enumerated(EnumType.STRING)
```

```
    private EnumHipoteca enumHipoteca;
    //Es el papeo logico que sirve para las listas dentro de una clases
    @ManyToOne
    @JoinColumn(name = "hipoteca_fk")
    private Banco banco;
    public Hipoteca() {

    }

    public Hipoteca(double monto, double interes, Calendar fechaInicio, Calendar fechaFin, double pagoMensualidad, Casa casa, int numeroPagos, Persona deudor, Persona garante, EnumHipoteca enumHipoteca, int pagosFaltantes) {
        this.monto = monto;
        this.interes = interes;
        this.fechaInicio = fechaInicio;
        this.fechaFin = fechaFin;
        this.pagoMensualidad = pagoMensualidad;
        this.casa = casa;
        this.numeroPagos = numeroPagos;
        this.deudor = deudor;
        this.garante = garante;
        this.enumHipoteca = enumHipoteca;
        this.pagosFaltantes= pagosFaltantes;
    }

    public int getPagosFaltantes() {
        return pagosFaltantes;
    }

    public void setPagosFaltantes(int pagosFaltantes) {
        this.pagosFaltantes = pagosFaltantes;
```

```
    }

    public Banco getBanco() {
        return banco;
    }

    public void setBanco(Banco banco) {
        this.banco = banco;
    }

    public int getCodigo() {
        return codigo;
    }

    public void setCodigo(int codigo) {
        this.codigo = codigo;
    }

    public double getMonto() {
        return monto;
    }

    public void setMonto(double monto) {
```

```
        this.monto = monto;
    }

    public double getInteres() {
        return interes;
    }

    public void setInteres(double interes) {
        this.interes = interes;
    }

    public Calendar getFechaInicio() {
        return fechaInicio;
    }

    public void setFechaInicio(Calendar fechaInicio) {
        this.fechaInicio = fechaInicio;
    }

    public Calendar getFechaFin() {
        return fechaFin;
    }

    public void setFechaFin(Calendar fechaFin) {
        this.fechaFin = fechaFin;
    }



    public double getPagoMensualidad() {
```

```
        return pagoMensualidad;
    }

    public void setPagoMensualidad(double pagoMensualidad) {
        this.pagoMensualidad = pagoMensualidad;
    }

    public Casa getCasa() {
        return casa;
    }

    public void setCasa(Casa casa) {
        this.casa = casa;
    }

    public int getNumeroPagos() {
        return numeroPagos;
    }

    public void setNumeroPagos(int numeroPagos) {
        this.numeroPagos = numeroPagos;
    }

    public Persona getDeudor() {
        return deudor;
    }

    public void setDeudor(Persona deudor) {
        this.deudor = deudor;
    }
```

```
    public Persona getGarante() {
        return garante;
    }

    public void setGarante(Persona garante) {
        this.garante = garante;
    }

    public EnumHipoteca getEnumHipoteca() {
        return enumHipoteca;
    }

    public void setEnumHipoteca(EnumHipoteca enumHipoteca) {
        this.enumHipoteca = enumHipoteca;
    }

    @Override
    public String toString() {
        return  "\ncodigo=" + codigo + "\n monto=" + monto + "\n interes=" + interes + "\n fechaInicio=" + fechaInicio + "\n fechaFin=" + fechaFin + "\n pagoMensualidad=" + pagoMensualidad + "\n casa=" + casa + "\n numeroPagos=" + numeroPagos + "\n pagosFaltantes=" + pagosFaltantes +  "\n garante=" + garante + "\n enumHipoteca=" + enumHipoteca ;
    }

    @Override
    public int hashCode() {
        int hash = 3;
```

```
        hash = 83 * hash + this.codigo;
        return hash;
    }

    @Override
    public boolean equals(Object obj) {
        if (this == obj) {
            return true;
        }
        if (obj == null) {
            return false;
        }
        if (getClass() != obj.getClass()) {
            return false;
        }
        final Hipoteca other = (Hipoteca) obj;
        if (this.codigo != other.codigo) {
            return false;
        }
        return true;
    }

}
```

Creacion de la clase Persona

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.modelo;
```

```
import java.io.Serializable;
import java.util.Calendar;
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.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;
import javax.persistence.Temporal;


/**
 *
 * @author ASUS
 */
@Entity
@Table(name="personas")
@NamedQueries({
    @NamedQuery(name = "Persona.findAll", query = "SELECT p FROM Persona p"),
    @NamedQuery(name = "Persona.findByCedula", query = "SELECT p FROM Persona p WHERE p.cedula = :cedula"),
    @NamedQuery(name = "Persona.findByCodigo", query = "SELECT p FROM Persona p WHERE p.codigo = :codigo"),
})
public class Persona implements Serializable{
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name ="codigo")
    private int codigo;
```

```
    @Column(name ="cedula")
    private String cedula;
    @Column(name ="nombre")
    private String nombre;
    @Column(name ="apellido")
    private String apellido;
     @Column(name ="direccion")
    private String direccion;
    @Column(name ="fecha_Nacimiento")
    @Temporal(javax.persistence.TemporalType.DATE)
    private Calendar fechaNacimiento;
    @Column(name="sueldo")
    private double sueldo;

    public Persona() {

    }

    public Persona(String cedula, String nombre, String apellido, String direccion, Calendar fechaNacimiento,
double sueldo) {
        this.cedula = cedula;
        this.nombre = nombre;
        this.apellido = apellido;
        this.direccion = direccion;
        this.fechaNacimiento = fechaNacimiento;
        this.sueldo = sueldo;
    }

    public double getSueldo() {
        return sueldo;
```

```
    }

    public void setSueldo(double sueldo) {
        this.sueldo = sueldo;
    }


    public String getApellido() {
        return apellido;
    }

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



    public int getCodigo() {
        return codigo;
    }

    public void setCodigo(int codigo) {
        this.codigo = codigo;
    }

    public String getCedula() {
        return cedula;
    }

    public void setCedula(String cedula) {
```

```
        this.cedula = cedula;
    }

    public String getNombre() {
        return nombre;
    }

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

    public String getDireccion() {
        return direccion;
    }

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

    public Calendar getFechaNacimiento() {
        return fechaNacimiento;
    }

    public void setFechaNacimiento(Calendar fechaNacimiento) {
        this.fechaNacimiento = fechaNacimiento;
    }

    @Override
    public String toString() {
        return "codigo=" + codigo + "\n" + nombre + " " + apellido +  "\n"+" Sueldo="+sueldo;
```

```
}



@Override
public int hashCode() {
    int hash = 5;
    hash = 53 * hash + this.codigo;
    return hash;
}

@Override
public boolean equals(Object obj) {
    if (this == obj) {
        return true;
    }
    if (obj == null) {
        return false;
    }
    if (getClass() != obj.getClass()) {
        return false;
    }
    final Persona other = (Persona) obj;
    if (this.codigo != other.codigo) {
        return false;
    }
    return true;
}
```

}

A continuacion se procedio a crear una carpeta de excepciones y controlador donde se podrá apreciar las expresiones regulares, asi como también esta controlado el numero de dígitos de la cedula que sea una real, haciendo referencia también en el código se controla espacio en blanco, fechas y características desarrolladas asi como también la programación genérica el MVC java 8, con el fin de cumplir cada requisito de problema planteado por el docente.

CedulaExepcion:

```
/*

 * To change this license header, choose License Headers in Project Properties.

 * To change this template file, choose Tools | Templates

 * and open the template in the editor.

 */

package ec.edu.ups.excepciones;



/**

 *

 * @author ASUS

 */

public class CedulaExcepcion extends Exception{

    public static String mensaje = "La cedula es invalida,Recuerde ingresar los 10 digitos sin guion";



    public CedulaExcepcion() {

        super(mensaje);

    }



}
```

Controlador genérico

```
/*

 * To change this license header, choose License Headers in Project Properties.

 * To change this template file, choose Tools | Templates

 * and open the template in the editor.

 */

package ec.edu.ups.controlador;
```

```
import java.util.ArrayList;
import java.util.List;
import javax.persistence.EntityManager;

/**
 *
 * @author ASUS
 * @param <E>
 */
public abstract class AbstractControlador<E> {

    private List<E> listado;
    private Class<E> tipo;
    private EntityManager em;

    public AbstractControlador(Class<E> tipo, EntityManager em) {
        listado = new ArrayList<>();
        this.em = em;
        this.tipo = tipo;
        this.listado = findAll();
    }

    public boolean crear(E obj) {
        em.getTransaction().begin();
        em.persist(obj);
        em.getTransaction().commit();
        this.listado = findAll();
        return true;
    }
```

```
    public E buscar(int id) {
        E resultado = em.find(tipo, id);
        return resultado;
    }

    public boolean eliminar(E obj) {
        em.getTransaction().begin();
        em.remove(obj);
        em.getTransaction().commit();
        this.listado = findAll();
        return true;
    }

    public boolean actualizar(E obj) {
        em.getTransaction().begin();
        em.merge(obj);
        em.getTransaction().commit();
        this.listado = findAll();
        return true;
    }

    public abstract List<E> findAll();

    public List<E> getListado() {
        return listado;
    }

    public void setListado(List<E> listado) {
        this.listado = listado;
    }
```

```
    public EntityManager getEm() {
        return em;
    }

    public void setEm(EntityManager em) {
        this.em = em;
    }

}
```

Controlador banco

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.controlador;

import java.util.ArrayList;
import java.util.List;
import javax.persistence.EntityManager;

/**
 *
 * @author ASUS
 * @param <E>
 */
public abstract class AbstractControlador<E> {

    private List<E> listado;
    private Class<E> tipo;
```

```
    private EntityManager em;

    public AbstractControlador(Class<E> tipo, EntityManager em) {
        listado = new ArrayList<>();
        this.em = em;
        this.tipo = tipo;
        this.listado = findAll();
    }

    public boolean crear(E obj) {
        em.getTransaction().begin();
        em.persist(obj);
        em.getTransaction().commit();
        this.listado = findAll();
        return true;
    }

    public E buscar(int id) {
        E resultado = em.find(tipo, id);
        return resultado;
    }

    public boolean eliminar(E obj) {
        em.getTransaction().begin();
        em.remove(obj);
        em.getTransaction().commit();
        this.listado = findAll();
        return true;
    }
```

```
    public boolean actualizar(E obj) {
        em.getTransaction().begin();
        em.merge(obj);
        em.getTransaction().commit();
        this.listado = findAll();
        return true;
    }

    public abstract List<E> findAll();

    public List<E> getListado() {
        return listado;
    }

    public void setListado(List<E> listado) {
        this.listado = listado;
    }

    public EntityManager getEm() {
        return em;
    }

    public void setEm(EntityManager em) {
        this.em = em;
    }

}
```

Para llevar a cabo el GUI se consideró pertinente la creación de 5 ventanas donde a continuación se podrá aprecias la principal llamada VtnPrincipal y a continuación se procederá a la explicación de la funcionalidad del mismo.

Ventana Principal.

```
/*
```

```
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.vista;

import ec.edu.ups.controlador.ControladorPrincipal;
import java.util.Locale;
import java.util.ResourceBundle;

/**
 *
 * @author ASUS
 */
public class VtnPrincipal extends javax.swing.JFrame {

    private ControladorPrincipal controladorPrincipal;



    /**
     *
     */
    public VentanaPersona ventanaPersona;

    /**
     * Creates new form VtnAdministrador
     */
    public VtnPrincipal() {
        controladorPrincipal = new ControladorPrincipal();
        initComponents();
```

```
        this.setLocationRelativeTo(null);



    }



    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jMenuItem1 = new javax.swing.JMenuItem();
        jMenuItem2 = new javax.swing.JMenuItem();
        dtkAdministrador = new javax.swing.JDesktopPane();
        infPrincipal = new javax.swing.JInternalFrame();
        jPanel1 = new javax.swing.JPanel();
        mnuAdministradorPrincipal = new javax.swing.JMenuBar();
        mnuBanco = new javax.swing.JMenu();
        mnuCasas = new javax.swing.JMenuItem();
        mnuPersonas = new javax.swing.JMenuItem();
        mnuHipotecas = new javax.swing.JMenuItem();
        mnuPagos = new javax.swing.JMenuItem();

        jMenuItem1.setText("jMenuItem1");
```

```
        jMenuItem2.setText("jMenuItem2");

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        infPrincipal.setBackground(new java.awt.Color(255, 255, 255));
        infPrincipal.setVisible(true);

        jPanel1.setBackground(new java.awt.Color(204, 204, 204));
        jPanel1.setPreferredSize(new java.awt.Dimension(989, 448));

        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 1084, Short.MAX_VALUE)
        );
        jPanel1Layout.setVerticalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 538, Short.MAX_VALUE)
        );

        javax.swing.GroupLayout infPrincipalLayout = new
javax.swing.GroupLayout(infPrincipal.getContentPane());
        infPrincipal.getContentPane().setLayout(infPrincipalLayout);
        infPrincipalLayout.setHorizontalGroup(
            infPrincipalLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 1084, Short.MAX_VALUE)
        );
        infPrincipalLayout.setVerticalGroup(
            infPrincipalLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
        .addGroup(infPrincipalLayout.createSequentialGroup()
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 618, Short.MAX_VALUE)
            .addContainerGap())
    );

    dtkAdministrador.setLayer(infPrincipal, javax.swing.JLayeredPane.DEFAULT_LAYER);

    javax.swing.GroupLayout dtkAdministradorLayout = new javax.swing.GroupLayout(dtkAdministrador);
    dtkAdministrador.setLayout(dtkAdministradorLayout);
    dtkAdministradorLayout.setHorizontalGroup(
        dtkAdministradorLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(infPrincipal)
    );
    dtkAdministradorLayout.setVerticalGroup(
        dtkAdministradorLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(infPrincipal)
    );

    mnuAdministradorPrincipal.setBackground(new java.awt.Color(255, 102, 51));
    mnuAdministradorPrincipal.setForeground(new java.awt.Color(255, 102, 0));

    mnuBanco.setText("Prestamo");

    mnuCasas.setText("Ver Casas");
    mnuCasas.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            mnuCasasActionPerformed(evt);
        }
    });
    mnuBanco.add(mnuCasas);
```

```
mnuPersonas.setText("Ver Personas");
mnuPersonas.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        mnuPersonasActionPerformed(evt);
    }
});
mnuBanco.add(mnuPersonas);

mnuHipotecas.setText("Ver Hipotecas");
mnuHipotecas.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        mnuHipotecasActionPerformed(evt);
    }
});
mnuBanco.add(mnuHipotecas);

mnuPagos.setText("Ver Pagos");
mnuPagos.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        mnuPagosActionPerformed(evt);
    }
});
mnuBanco.add(mnuPagos);

mnuAdministradorPrincipal.add(mnuBanco);

setJMenuBar(mnuAdministradorPrincipal);

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

```
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(dtkAdministrador)
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(dtkAdministrador)
        );

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

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

        VentanaHipotecas ventanaHipotecas = new VentanaHipotecas(controladorPrincipal);
        infPrincipal.setTitle("Hipotecas");
        infPrincipal.setContentPane(ventanaHipotecas);
        infPrincipal.setVisible(true);
        dtkAdministrador.moveToFront(infPrincipal);

    }

    private void mnuCasasActionPerformed(java.awt.event.ActionEvent evt) {
        VentanaCasas ventanaCasas = new VentanaCasas(controladorPrincipal);
        infPrincipal.setTitle("Casas");
        infPrincipal.setContentPane(ventanaCasas);
        infPrincipal.setVisible(true);
        dtkAdministrador.moveToFront(infPrincipal);
    }
```

```
private void mnuPersonasActionPerformed(java.awt.event.ActionEvent evt) {
    ventanaPersona = new VentanaPersona(controladorPrincipal);
    infPrincipal.setTitle("Personas");
    infPrincipal.setContentPane(ventanaPersona);
    infPrincipal.setVisible(true);
    dtkAdministrador.moveToFront(infPrincipal);



}

private void mnuPagosActionPerformed(java.awt.event.ActionEvent evt) {
   VentanaPagos ventanaPagos= new VentanaPagos(controladorPrincipal);
    infPrincipal.setTitle("Pagos");
    infPrincipal.setContentPane(ventanaPagos);
    infPrincipal.setVisible(true);
    dtkAdministrador.moveToFront(infPrincipal);
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
     */
    try {
```

```
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
java.util.logging.Logger.getLogger(VtnPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (InstantiationException ex) {
java.util.logging.Logger.getLogger(VtnPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (IllegalAccessException ex) {
java.util.logging.Logger.getLogger(VtnPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
java.util.logging.Logger.getLogger(VtnPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
        }
        //</editor-fold>
        //</editor-fold>
        //</editor-fold>
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new VtnPrincipal().setVisible(true);
```

```
            }
        });
    }



    // Variables declaration - do not modify
    private javax.swing.JDesktopPane dtkAdministrador;
    private javax.swing.JInternalFrame infPrincipal;
    private javax.swing.JMenuItem jMenuItem1;
    private javax.swing.JMenuItem jMenuItem2;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JMenuBar mnuAdministradorPrincipal;
    private javax.swing.JMenu mnuBanco;
    private javax.swing.JMenuItem mnuCasas;
    private javax.swing.JMenuItem mnuHipotecas;
    private javax.swing.JMenuItem mnuPagos;
    private javax.swing.JMenuItem mnuPersonas;
    // End of variables declaration
}
```

ventana persona

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.vista;


import ec.edu.ups.controlador.ControladorPrincipal;
import ec.edu.ups.excepciones.CedulaExcepcion;
import ec.edu.ups.excepciones.VacioExcepcion;
```

```
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.GregorianCalendar;
import java.util.Locale;
import java.util.ResourceBundle;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;
import javax.swing.table.DefaultTableModel;
import ec.edu.ups.modelo.Persona;

/**
 *
 * @author ASUS
 */
public class VentanaPersona extends javax.swing.JPanel {

    private ControladorPrincipal controladorPrincipal;
    private DefaultTableModel defaultTableModel;
    private SimpleDateFormat formato;

    /**
     * Creates new form VentanaPersona
     *
     * @param controladorPrincipal
     */
```

```
    public VentanaPersona(ControladorPrincipal controladorPrincipal) {
        this.controladorPrincipal = controladorPrincipal;
        initComponents();
        defaultTableModel = (DefaultTableModel) TablaPersonas.getModel();
        formato = new SimpleDateFormat("dd/mm/yyyy");
        this.cargarDatosTabla();
        this.TablaPersonas.getSelectionModel().addListSelectionListener(new ListSelectionListener() {
            @Override
            public void valueChanged(ListSelectionEvent e) {
                vizualizarDatos(TablaPersonas.getSelectedRow());
            }
        });
    }

    public void cargarDatosTabla() {
        defaultTableModel.setRowCount(0);

        for (Persona cliente : controladorPrincipal.getPersonaDAO().getListado()) {
            Calendar fecha = cliente.getFechaNacimiento();
            String dia = Integer.toString(fecha.get(Calendar.DAY_OF_MONTH));
            String mes = Integer.toString(fecha.get(Calendar.MONTH));
            String año = Integer.toString(fecha.get(Calendar.YEAR));
            String fechaNacimiento = dia + "/" + mes + "/" + año;
            String datos[] = {String.valueOf(cliente.getCodigo()),cliente.getCedula(),cliente.getNombre(),cliente.getApellido(),cliente.getDireccion(),fechaNacimiento,String.valueOf(cliente.getSueldo())};
            defaultTableModel.addRow(datos);
        }

    }
```

```
public void vizualizarDatos(int posicion) {
    if (posicion >= 0) {
        Persona persona = this.controladorPrincipal.getPersonaDAO().getListado().get(posicion);
        Calendar fecha = persona.getFechaNacimiento();
        String dia = Integer.toString(fecha.get(Calendar.DAY_OF_MONTH));
        String mes = Integer.toString(fecha.get(Calendar.MONTH));
        String año = Integer.toString(fecha.get(Calendar.YEAR));
        String fechaNacimiento = dia + "/" + mes + "/" + año;
        txtCodigo.setText(String.valueOf(persona.getCodigo()));
        txtNombre.setText(persona.getNombre());
        txtCedula.setText(persona.getCedula());
        txtDireccion.setText(persona.getDireccion());
        txtSueldo.setText(String.valueOf(persona.getSueldo()));
        txtApellido.setText(persona.getApellido());
         txtFechaNacimiento.setText(fechaNacimiento);
        txtCedula.setEditable(false);

    } else {
        txtCodigo.setText("");
        txtNombre.setText("");
        txtCedula.setText("");
        txtDireccion.setText("");
        txtSueldo.setText("");
        txtApellido.setText("");
        txtFechaNacimiento.setText("");

    }
}

/**
```

```
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();
        jPanel1 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        txtCodigo = new javax.swing.JTextField();
        txtCedula = new javax.swing.JTextField();
        txtNombre = new javax.swing.JTextField();
        txtApellido = new javax.swing.JTextField();
        jLabel6 = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();
        txtDireccion = new javax.swing.JTextField();
        txtFechaNacimiento = new javax.swing.JTextField();
        jPanel2 = new javax.swing.JPanel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jLabel8 = new javax.swing.JLabel();
        txtSueldo = new javax.swing.JTextField();
```

```
jScrollPane2 = new javax.swing.JScrollPane();
TablaPersonas = new javax.swing.JTable();

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "Title 1", "Title 2", "Title 3", "Title 4"
    }
));
jScrollPane1.setViewportView(jTable1);

jLabel1.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
jLabel1.setText("Personas");

jLabel2.setText("ID");

jLabel3.setText("Cedula");

jLabel4.setText("Apellido");

jLabel5.setText("Nombre");

txtCodigo.setEditable(false);

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

```
        jLabel7.setText("Fecha Nacimiento");

        jButton1.setText("OK");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        jButton2.setText("Eliminar");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });

        jButton3.setText("Limpiar");
        jButton3.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton3ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
        jPanel2.setLayout(jPanel2Layout);
        jPanel2Layout.setHorizontalGroup(
            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jButton1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, 195, Short.MAX_VALUE)
            .addComponent(jButton3, javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        );
        jPanel2Layout.setVerticalGroup(
            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addComponent(jButton1, javax.swing.GroupLayout.PREFERRED_SIZE, 64,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 19,
Short.MAX_VALUE)
                .addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jButton3, javax.swing.GroupLayout.PREFERRED_SIZE, 72,
javax.swing.GroupLayout.PREFERRED_SIZE))
        );

        jLabel8.setText("Sueldo ");

        TablaPersonas.setModel(new javax.swing.table.DefaultTableModel(
            new Object [][] {
                {null, null, null, null, null, null, null},
                {null, null, null, null, null, null, null},
                {null, null, null, null, null, null, null},
                {null, null, null, null, null, null, null}
            },
            new String [] {
                "Id", "Cedula", "Nombre", "Apellido", "Direccion", "Fecha Nacimiento", "Sueldo"
            }
        ) {
```

```
            boolean[] canEdit = new boolean [] {
                false, false, false, false, false, false, false
            };

            public boolean isCellEditable(int rowIndex, int columnIndex) {
                return canEdit [columnIndex];
            }
        });
        jScrollPane2.setViewportView(TablaPersonas);
        if (TablaPersonas.getColumnModel().getColumnCount() > 0) {
            TablaPersonas.getColumnModel().getColumn(0).setPreferredWidth(40);
            TablaPersonas.getColumnModel().getColumn(1).setPreferredWidth(55);
            TablaPersonas.getColumnModel().getColumn(6).setPreferredWidth(55);
        }

        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(318, 318, 318)
                        .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 141,
javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(30, 30, 30)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addComponent(jScrollPane2)
                            .addGroup(jPanel1Layout.createSequentialGroup()
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addComponent(jLabel2)
                        .addComponent(jLabel4)
                        .addComponent(jLabel5)
                        .addComponent(jLabel3)
                        .addComponent(jLabel6)
                        .addComponent(jLabel7)
                        .addComponent(jLabel8))
                    .addGap(27, 27, 27)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                        .addComponent(txtCodigo)
                        .addComponent(txtCedula)
                        .addComponent(txtNombre)
                        .addComponent(txtApellido)
                        .addComponent(txtDireccion)
                        .addComponent(txtFechaNacimiento)
                        .addComponent(txtSueldo, javax.swing.GroupLayout.DEFAULT_SIZE, 201,
Short.MAX_VALUE))
                    .addGap(18, 18, 18)
                    .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))))
            .addContainerGap())
    );
    jPanel1Layout.setVerticalGroup(
        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
                .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()

                    .addContainerGap()

                    .addComponent(jLabel1,          javax.swing.GroupLayout.PREFERRED_SIZE,          24,
javax.swing.GroupLayout.PREFERRED_SIZE)

                    .addGap(18, 18, 18)


.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                        .addComponent(jLabel2)

                        .addComponent(txtCodigo,                javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addGap(21, 21, 21)


.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                        .addComponent(jLabel3)

                        .addComponent(txtCedula,                javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addGap(18, 18, 18)


.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                        .addComponent(jLabel5,          javax.swing.GroupLayout.PREFERRED_SIZE,          22,
javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(txtNombre,                javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)


.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                        .addComponent(jLabel4,                  javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 22, javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(txtApellido,              javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE,          javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

                    .addGap(18, 18, 18)


.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
```

```
                    .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                        .addComponent(jLabel6)
                        .addComponent(txtDireccion, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGap(18, 18, 18)
                    .addComponent(jLabel7))
                .addComponent(txtFechaNacimiento, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel8)
                    .addComponent(txtSueldo, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, 109, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(328, 328, 328))
        );


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

```
        );
    }// </editor-fold>

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

        try {
            Date fecha1 = formato.parse(txtFechaNacimiento.getText());
            Calendar fecha = new GregorianCalendar();

            fecha.setTime(fecha1);

            try {
                if (txtCodigo.getText().equals("")) {
                    String mensaje = controladorPrincipal.crearPersona(txtCedula.getText(), txtNombre.getText(),
txtApellido.getText(), txtDireccion.getText(), fecha, Double.valueOf(txtSueldo.getText()));
                    JOptionPane.showMessageDialog(this, mensaje);

                } else {
                    String mensaje = controladorPrincipal.actualizarPersona(Integer.parseInt(txtCodigo.getText()),
txtCedula.getText(), txtNombre.getText(), txtApellido.getText(), txtDireccion.getText(), fecha,
Double.valueOf(txtSueldo.getText()));
                    JOptionPane.showMessageDialog(this, mensaje);
                }
            } catch (VacioExcepcion | CedulaExcepcion ex) {
                JOptionPane.showMessageDialog(this, ex.toString());
            } finally {
                cargarDatosTabla();
                this.vizualizarDatos(-1);
            }
        } catch (ParseException ex) {
            JOptionPane.showMessageDialog(this, "Error en la fecha");
```

```
    }
}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    txtCodigo.setText("");
    txtNombre.setText("");
    txtCedula.setText("");
    txtDireccion.setText("");
    txtSueldo.setText("");
    txtApellido.setText("");
    txtCedula.setEditable(true);
    txtFechaNacimiento.setText("");
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        Persona persona=controladorPrincipal.BuscarPersona(txtCedula.getText());
        controladorPrincipal.getPersonaDAO().eliminar(persona);
        cargarDatosTabla();
           this.vizualizarDatos(-1);
           JOptionPane.showMessageDialog(this, "Exitoso");
    } catch (Exception e) {
         JOptionPane.showMessageDialog(this, "Error");
    }
}


// Variables declaration - do not modify
private javax.swing.JTable TablaPersonas;
private javax.swing.JButton jButton1;
```

```
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton3;
    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.JLabel jLabel8;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JScrollPane jScrollPane2;
    private javax.swing.JTable jTable1;
    private javax.swing.JTextField txtApellido;
    private javax.swing.JTextField txtCedula;
    private javax.swing.JTextField txtCodigo;
    private javax.swing.JTextField txtDireccion;
    private javax.swing.JTextField txtFechaNacimiento;
    private javax.swing.JTextField txtNombre;
    private javax.swing.JTextField txtSueldo;
    // End of variables declaration


}
```

Ventana hipotecas

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

```
package ec.edu.ups.vista;

import ec.edu.ups.controlador.ControladorPrincipal;
import ec.edu.ups.excepciones.GaranteExepcion;
import ec.edu.ups.excepciones.HipotecaExcepcion;
import ec.edu.ups.excepciones.VacioExcepcion;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.GregorianCalendar;
import javax.swing.JOptionPane;
import javax.swing.event.ListSelectionEvent;
import javax.swing.event.ListSelectionListener;
import javax.swing.table.DefaultTableModel;
import ec.edu.ups.modelo.Casa;
import ec.edu.ups.modelo.EnumHipoteca;
import ec.edu.ups.modelo.Hipoteca;
import ec.edu.ups.modelo.Persona;

/**
 *
 * @author ASUS
 */
public class VentanaHipotecas extends javax.swing.JPanel {

    private SimpleDateFormat formato;
    private SimpleDateFormat formato1;
    private ControladorPrincipal controladorPrincipal;
    private DefaultTableModel tabla;
```

```
    /**
     * Creates new form VentanaHipotecas
     *
     * @param controladorPrincipal
     */
    public VentanaHipotecas(ControladorPrincipal controladorPrincipal) {
        this.controladorPrincipal = controladorPrincipal;
        formato = new SimpleDateFormat("dd/mm/yyyy");
        formato1 = new SimpleDateFormat("dd/mm/yyyy");
//        tabla = (DefaultTableModel) tablaFechas.getModel();
        initComponents();
        txtCedulaGarante.setEnabled(false);
        txtInteres.setText("8.99");
        txtSINO.setText("NO");
        //this.cargarDatosTabla();
    }

    public void cargarDatosTabla()  {
        tabla = (DefaultTableModel) tablaFechas.getModel();
        try {
           Date fecha1 = formato.parse(txtFechaInicio.getText());
        Calendar fecha = new GregorianCalendar();
        fecha.setTime(fecha1);
         int n = Integer.valueOf(txtNumeroPagos.getText());
        tabla.setRowCount(0);
        int contador = 0;
            String dia = Integer.toString(fecha.get(Calendar.DAY_OF_MONTH));
            String mes = Integer.toString(fecha.get(Calendar.MONTH));
            String año = Integer.toString(fecha.get(Calendar.YEAR));
```

```
                int m = Integer.valueOf(mes)+1;
                mes = String.valueOf(m);
            for (int i = 0; i < n; i++) {
                contador += 1;
                 m = Integer.valueOf(mes)+1;
                mes = String.valueOf(m);
                if (m>12) {
                    int a = Integer.valueOf(año)+1;
                    año = String.valueOf(a);
                    mes="1";
                } else {
                }
                String fechaPago = dia + "/" + mes + "/" + año;
                String datos[] = {fechaPago, String.valueOf(contador)};
                tabla.addRow(datos);
            }


            } catch (Exception e) {
            }


        }

        public void vizualizarDatos(int posicion) {
            if (posicion >= 0) {
                Hipoteca hipoteca = this.controladorPrincipal.getHipotecaDAO().getListado().get(posicion);
                String deudor = hipoteca.getDeudor().getCedula();
                String garante = "";
                if (hipoteca.getGarante() != null) {
```

```
            garante = hipoteca.getGarante().getCedula();
            txtSINO.setText("SI");
        } else {
            txtSINO.setText("NO");
        }
        /*
        if (null != hipoteca.getEnumHipoteca()) {
            switch (hipoteca.getEnumHipoteca()) {
                case NORMAL:
                    cmbEstado.setSelectedIndex(0);
                    break;
                case PAGADO:
                    cmbEstado.setSelectedIndex(1);
                    break;
                case EMBARGADO:
                    cmbEstado.setSelectedIndex(2);
                    break;
                default:
                    break;
            }
        }
         */
        String casa = String.valueOf(hipoteca.getCasa().getCodigo());
        txtCodigo.setText(String.valueOf(hipoteca.getCodigo()));
        txtCasa.setText(casa);
        txtDeudor.setText(deudor);
        txtCedulaGarante.setText(garante);
        txtMonto.setText(String.valueOf(hipoteca.getMonto()));
        txtInteres.setText(String.valueOf(hipoteca.getInteres()));
        Calendar fecha = hipoteca.getFechaInicio();
```

```
            String dia = Integer.toString(fecha.get(Calendar.DAY_OF_MONTH));
            String mes = Integer.toString(fecha.get(Calendar.MONTH));
            String año = Integer.toString(fecha.get(Calendar.YEAR));
            String fechaInicio = dia + "/" + mes + "/" + año;
            txtFechaInicio.setText(fechaInicio);
            Calendar fecha1 = hipoteca.getFechaFin();
            String dia1 = Integer.toString(fecha.get(Calendar.DAY_OF_MONTH));
            String mes1 = Integer.toString(fecha.get(Calendar.MONTH));
            String año1 = Integer.toString(fecha.get(Calendar.YEAR));
            String fechaFin = dia + "/" + mes + "/" + año;
            txtFechaInicio.setText(fechaInicio);
            txtFechaInicio.setText(fechaFin);

            probarHipoteca();

        } else {
            txtCasa.setText("");
            txtCedulaGarante.setText("");
            txtCodigo.setText("");
            txtDatosCasa.setText("");
            txtDeudor.setText("");
            txtInteres.setText("");
            txtMonto.setText("");
            txtNumeroPagos.setText("");
            txtNumeroPagos.setText("");
            txtPagoMensual.setText("");
            txtSINO.setText("");
            txtFechaInicio.setText("");
            txtFechaInicio.setText("");
```

```
    }
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jScrollPane3 = new javax.swing.JScrollPane();
    jTable1 = new javax.swing.JTable();
    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = 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();
    txtCodigo = new javax.swing.JTextField();
    txtInteres = new javax.swing.JTextField();
    txtMonto = new javax.swing.JTextField();
    txtFechaInicio = new javax.swing.JTextField();
    txtFechaFin = new javax.swing.JTextField();
    txtCasa = new javax.swing.JTextField();
    txtDeudor = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
```

```
jScrollPane1 = new javax.swing.JScrollPane();
txtDatosPersona = new javax.swing.JTextArea();
jScrollPane2 = new javax.swing.JScrollPane();
txtDatosCasa = new javax.swing.JTextArea();
jLabel8 = new javax.swing.JLabel();
jLabel9 = new javax.swing.JLabel();
jPanel2 = new javax.swing.JPanel();
jButton2 = new javax.swing.JButton();
jButton3 = new javax.swing.JButton();
jButton4 = new javax.swing.JButton();
jLabel10 = new javax.swing.JLabel();
jLabel11 = new javax.swing.JLabel();
txtSINO = new javax.swing.JTextField();
txtCedulaGarante = new javax.swing.JTextField();
jLabel12 = new javax.swing.JLabel();
jLabel13 = new javax.swing.JLabel();
jLabel14 = new javax.swing.JLabel();
txtNumeroPagos = new javax.swing.JTextField();
txtTotal = new javax.swing.JTextField();
txtPagoMensual = new javax.swing.JTextField();
jScrollPane5 = new javax.swing.JScrollPane();
tablaFechas = new javax.swing.JTable();

jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
```

```
            new String [] {
                "Title 1", "Title 2", "Title 3", "Title 4"
            }
        ));
        jScrollPane3.setViewportView(jTable1);

        jLabel1.setText("Id ");

        jLabel2.setText("Cedula ");

        jLabel3.setText("Id Casa ");

        jLabel4.setText("Valor  Casa");

        jLabel5.setText("Interes ");

        jLabel6.setText("Fecha inicio ");

        jLabel7.setText("Fecha fin ");

        txtCodigo.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                txtCodigoActionPerformed(evt);
            }
        });

        jButton1.setText("Calcular");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
```

```
    }
});

txtDatosPersona.setColumns(20);
txtDatosPersona.setRows(5);
jScrollPane1.setViewportView(txtDatosPersona);

txtDatosCasa.setColumns(20);
txtDatosCasa.setRows(5);
jScrollPane2.setViewportView(txtDatosCasa);

jLabel8.setText("Persona ");

jLabel9.setText("Casa ");

jButton2.setText("OK");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

jButton3.setText("Eliminar");
jButton3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton3ActionPerformed(evt);
    }
});

jButton4.setText("Limpiar");
```

```
        jButton4.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton4ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
        jPanel2.setLayout(jPanel2Layout);
        jPanel2Layout.setHorizontalGroup(
            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jButton2, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jButton3, javax.swing.GroupLayout.DEFAULT_SIZE, 195, Short.MAX_VALUE)
            .addComponent(jButton4, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );
        jPanel2Layout.setVerticalGroup(
            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 43, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jButton3, javax.swing.GroupLayout.PREFERRED_SIZE, 48, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jButton4, javax.swing.GroupLayout.PREFERRED_SIZE, 45, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(52, 52, 52))
        );

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

```
jLabel11.setText("Cedula Garante ");

txtSINO.setEditable(false);

jLabel12.setText("N Pagos ");

jLabel13.setText("Pagos Mensuales ");

jLabel14.setText("Pago Total ");

txtNumeroPagos.setEditable(false);

txtTotal.setEditable(false);

txtPagoMensual.setEditable(false);

tablaFechas.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null},
        {null, null},
        {null, null},
        {null, null}
    },
    new String [] {
        "Fecha", "Numero de pago"
    }
));
jScrollPane5.setViewportView(tablaFechas);
```

```
        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                    .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                        .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addGroup(jPanel1Layout.createSequentialGroup()
                                .addGap(8, 8, 8)
                                .addComponent(jLabel1))
                            .addComponent(jLabel7))
                        .addGap(28, 28, 28)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                            .addComponent(txtDeudor, javax.swing.GroupLayout.PREFERRED_SIZE, 121,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(txtFechaFin, javax.swing.GroupLayout.PREFERRED_SIZE, 123,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addGroup(jPanel1Layout.createSequentialGroup()
                                .addComponent(txtCodigo, javax.swing.GroupLayout.PREFERRED_SIZE, 121,
javax.swing.GroupLayout.PREFERRED_SIZE)
                                .addGap(67, 67, 67)
                                .addComponent(jLabel8))))
                        .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
                    .addComponent(jLabel2)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()
                            .addComponent(jLabel3)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(txtCasa, javax.swing.GroupLayout.PREFERRED_SIZE, 121,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()
                            .addComponent(jLabel4)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(txtMonto, javax.swing.GroupLayout.PREFERRED_SIZE, 121,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()
                            .addComponent(jLabel5)
                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(txtInteres, javax.swing.GroupLayout.PREFERRED_SIZE, 121,
javax.swing.GroupLayout.PREFERRED_SIZE))
                        .addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()
                            .addComponent(jLabel6)
                            .addGap(18, 18, 18)
                            .addComponent(txtFechaInicio, javax.swing.GroupLayout.PREFERRED_SIZE,
121, javax.swing.GroupLayout.PREFERRED_SIZE))))
                    .addGap(28, 28, 28)
                    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(18, 18, 18)
                        .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                        .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(64, 64, 64)
                        .addComponent(jLabel9))))
                .addGroup(javax.swing.GroupLayout.Alignment.LEADING, jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(jLabel10)
                        .addGap(18, 18, 18)
                        .addComponent(txtSINO, javax.swing.GroupLayout.PREFERRED_SIZE, 115, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(27, 27, 27)
                        .addComponent(jLabel11)
                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                        .addComponent(txtCedulaGarante, javax.swing.GroupLayout.PREFERRED_SIZE, 115, javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(jLabel13)
                        .addGap(18, 18, 18)
                        .addComponent(txtPagoMensual))
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(jLabel14)
                        .addGap(47, 47, 47)
                        .addComponent(txtTotal))
```

```
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(jLabel12)
                        .addGap(60, 60, 60)
                        .addComponent(txtNumeroPagos, javax.swing.GroupLayout.PREFERRED_SIZE, 299,
javax.swing.GroupLayout.PREFERRED_SIZE)))
                .addGap(18, 18, 18)
                .addComponent(jScrollPane5, javax.swing.GroupLayout.PREFERRED_SIZE, 268,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addComponent(jButton1, javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE, 1086, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(182, Short.MAX_VALUE))
    );
    jPanel1Layout.setVerticalGroup(
      jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(jPanel1Layout.createSequentialGroup()
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
          .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
              .addComponent(jLabel1)
              .addComponent(txtCodigo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
              .addComponent(jLabel2)
              .addComponent(txtDeudor, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
                .addComponent(jLabel3)

                .addComponent(txtCasa, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

                .addComponent(jLabel4)

                .addComponent(txtMonto, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

                .addComponent(jLabel5)

                .addComponent(txtInteres, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                .addComponent(jLabel6)

                .addComponent(txtFechaInicio, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                .addComponent(jLabel7)

                .addComponent(txtFechaFin, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

            .addGroup(jPanel1Layout.createSequentialGroup()

                .addGap(4, 4, 4)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                .addComponent(jLabel8)

                .addComponent(jLabel9))

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jScrollPane1)
                    .addComponent(jScrollPane2)
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE, 164, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(0, 0, Short.MAX_VALUE)))))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(jButton1)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
              .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel10)
                    .addComponent(jLabel11)
                    .addComponent(txtSINO, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(txtCedulaGarante, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(27, 27, 27)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel12)
                    .addComponent(txtNumeroPagos, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(jLabel13)
                    .addComponent(txtPagoMensual, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel14)

                    .addComponent(txtTotal, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))

                .addComponent(jScrollPane5, javax.swing.GroupLayout.PREFERRED_SIZE, 0,
Short.MAX_VALUE))

            .addContainerGap(441, Short.MAX_VALUE))

        );


        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(this);

        this.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)

        );

    }// </editor-fold>


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

        // TODO add your handling code here:

    }


    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

        try {

            Date fecha1 = formato.parse(txtFechaInicio.getText());
```

```
            Calendar fechaInicio = new GregorianCalendar();
            fechaInicio.setTime(fecha1);
            Date fecha2 = formato1.parse(txtFechaFin.getText());
            Calendar fechaFin = new GregorianCalendar();
            fechaFin.setTime(fecha2);
            Casa casa = controladorPrincipal.BuscarCasa(Integer.valueOf(txtCasa.getText()));
            EnumHipoteca estado = EnumHipoteca.NORMAL;
            estado = EnumHipoteca.NORMAL;

            Persona deudor = new Persona();
            try {
               deudor = controladorPrincipal.BuscarPersona(txtDeudor.getText());
            } catch (Exception ex) {
               JOptionPane.showMessageDialog(null, "Persona no encontrada");
            }

            try {
               if (txtCodigo.getText().equals("") && txtSINO.getText().equals("NO")) {

                  String mensaje = controladorPrincipal.crearHipoteca(Double.valueOf(txtMonto.getText()),
Double.valueOf(txtInteres.getText()), fechaInicio, fechaFin, Double.valueOf(txtPagoMensual.getText()),
casa, Integer.valueOf(txtNumeroPagos.getText()), deudor, null,
estado,Integer.valueOf(txtNumeroPagos.getText()));
                  JOptionPane.showMessageDialog(this, mensaje);
                  if (estado.equals(estado.PAGADO)) {
                     controladorPrincipal.actualizarEstado(false, casa.getCodigo());
                  } else {
                     controladorPrincipal.actualizarEstado(true, casa.getCodigo());
                  }

               } else if (txtCodigo.getText().equals("") && txtSINO.getText().equals("SI")) {
```

```
                Persona garante = controladorPrincipal.BuscarPersona(txtCedulaGarante.getText());

                String mensaje = controladorPrincipal.crearHipoteca(Double.valueOf(txtMonto.getText()),
Double.valueOf(txtInteres.getText()), fechaInicio, fechaFin, Double.valueOf(txtPagoMensual.getText()),
casa, Integer.valueOf(txtNumeroPagos.getText()), deudor, garante,
estado,Integer.valueOf(txtNumeroPagos.getText()));

                JOptionPane.showMessageDialog(this, mensaje);

                if (estado.PAGADO.equals(EnumHipoteca.PAGADO)) {

                    controladorPrincipal.actualizarEstado(false, casa.getCodigo());

                } else {

                    controladorPrincipal.actualizarEstado(true, casa.getCodigo());

                }


            } else {


                if (txtSINO.getText().equals("NO")) {

                    String mensaje =
controladorPrincipal.actualizarHipoteca(Integer.parseInt(txtCodigo.getText()),
Double.valueOf(txtMonto.getText()), Double.valueOf(txtInteres.getText()), fechaInicio, fechaFin,
Double.valueOf(txtPagoMensual.getText()), casa, Integer.valueOf(txtNumeroPagos.getText()), deudor,
null, estado);

                    JOptionPane.showMessageDialog(this, mensaje);

                    if (estado.PAGADO.equals(EnumHipoteca.PAGADO)) {

                        controladorPrincipal.actualizarEstado(false, casa.getCodigo());

                    } else {

                        controladorPrincipal.actualizarEstado(true, casa.getCodigo());

                    }


                } else if (txtSINO.getText().equals("SI")) {

                    Persona garante = controladorPrincipal.BuscarPersona(txtCedulaGarante.getText());

                    String mensaje =
controladorPrincipal.actualizarHipoteca(Integer.parseInt(txtCodigo.getText()),
Double.valueOf(txtMonto.getText()), Double.valueOf(txtInteres.getText()), fechaInicio, fechaFin,
Double.valueOf(txtPagoMensual.getText()), casa, Integer.valueOf(txtNumeroPagos.getText()), deudor,
garante, estado);
```

```
                    JOptionPane.showMessageDialog(this, mensaje);
                    if (estado.equals(EnumHipoteca.PAGADO)) {
                        controladorPrincipal.actualizarEstado(false, casa.getCodigo());
                    } else {
                        controladorPrincipal.actualizarEstado(true, casa.getCodigo());
                    }

                }
            }
        } catch (VacioExcepcion | GaranteExepcion | HipotecaExcepcion ex) {
            JOptionPane.showMessageDialog(this, ex.toString());
        } finally {
          //  cargarDatosTabla();
            this.vizualizarDatos(-1);
        }
    } catch (Exception ex) {
        JOptionPane.showMessageDialog(null, "Eroor: Pruebe Primero con el boton Probar Hipoteca ");
    }
}
public void probarHipoteca() {
    try {

        Date fecha1 = formato.parse(txtFechaInicio.getText());
        Calendar fechaInicio = new GregorianCalendar();
        fechaInicio.setTime(fecha1);
        Date fecha2 = formato1.parse(txtFechaFin.getText());
        Calendar fechaFin = new GregorianCalendar();
        fechaFin.setTime(fecha2);
        int diaInicio = fechaInicio.get(Calendar.DAY_OF_YEAR);
        int diaFin = fechaFin.get(Calendar.DAY_OF_YEAR);
```

```
            int mesInicio = fechaInicio.get(Calendar.MONTH);
            int añoInicio = fechaInicio.get(Calendar.YEAR);
            int mesFin = fechaFin.get(Calendar.MONTH);
            int añoFin = fechaFin.get(Calendar.YEAR);
            if (fechaInicio == null || fechaFin == null) {
                JOptionPane.showMessageDialog(null, "Colocar Fechas ");
            } else if (añoFin - añoInicio < 0) {
                JOptionPane.showMessageDialog(null, "Fechas mal Colocadas ");
            } else if (añoFin - añoInicio == 0 && mesFin - mesInicio < 0) {
                JOptionPane.showMessageDialog(null, "Fechas mal Colocadas ");
            } else if (añoFin - añoInicio == 0 && diaFin - diaInicio < 0) {
                JOptionPane.showMessageDialog(null, "Fechas mal Colocadas ");
            } else {
                double valorTotal = Double.valueOf(txtMonto.getText()) + (Double.valueOf(txtMonto.getText()) *
(Double.valueOf(txtInteres.getText()) / 100));

                if ((añoInicio - añoFin) == 0) {
                    int valor = mesFin - mesInicio;
                    if (valor == 0) {
                        txtNumeroPagos.setText("1");

                    } else {
                        txtNumeroPagos.setText(String.valueOf(valor));
                    }

                } else if ((añoFin - añoInicio) == 1) {
                    int valor = (12 - mesFin + 1) + (12 - mesInicio + 1);
                    txtNumeroPagos.setText(String.valueOf(valor));
                } else if ((añoFin - añoInicio) == 2) {
                    int valor = mesFin + (12 - mesInicio) + 12;
```

```
                txtNumeroPagos.setText(String.valueOf(valor));
            } else {
                int años = añoFin - añoInicio - 2;
                int valor = mesFin - (12 - mesInicio + 1) + (años * 12);
                txtNumeroPagos.setText(String.valueOf(valor));
            }
            txtPagoMensual.setText(String.valueOf(Double.valueOf(txtMonto.getText()) / Double.valueOf(txtNumeroPagos.getText())));
            Persona persona = controladorPrincipal.BuscarPersona(txtDeudor.getText());
            txtDatosPersona.setText(persona.toString());

            Casa casa = controladorPrincipal.BuscarCasa(Integer.valueOf(txtCasa.getText()));
            txtDatosCasa.setText(casa.toString());
            if (persona.getSueldo() < Double.valueOf(txtPagoMensual.getText()) && casa.getAvaluo() < valorTotal) {
                txtSINO.setText("SI");
                txtCedulaGarante.setEnabled(true);
            } else {
                txtSINO.setText("NO");
                txtCedulaGarante.setEnabled(false);
            }
            txtTotal.setText(String.valueOf(valorTotal));
        }
    } catch (Exception ex) {
        JOptionPane.showMessageDialog(null, "Error: Datos Faltantes o mal ingresados");
    }

}
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

}
```

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {


}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        probarHipoteca();
        cargarDatosTabla();
    } catch (Exception e) {
    }

}



// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
private javax.swing.JLabel jLabel14;
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.JLabel jLabel9;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JScrollPane jScrollPane2;
    private javax.swing.JScrollPane jScrollPane3;
    private javax.swing.JScrollPane jScrollPane5;
    private javax.swing.JTable jTable1;
    private javax.swing.JTable tablaFechas;
    private javax.swing.JTextField txtCasa;
    private javax.swing.JTextField txtCedulaGarante;
    private javax.swing.JTextField txtCodigo;
    private javax.swing.JTextArea txtDatosCasa;
    private javax.swing.JTextArea txtDatosPersona;
    private javax.swing.JTextField txtDeudor;
    private javax.swing.JTextField txtFechaFin;
    private javax.swing.JTextField txtFechaInicio;
    private javax.swing.JTextField txtInteres;
    private javax.swing.JTextField txtMonto;
    private javax.swing.JTextField txtNumeroPagos;
    private javax.swing.JTextField txtPagoMensual;
    private javax.swing.JTextField txtSINO;
    private javax.swing.JTextField txtTotal;
    // End of variables declaration
}
/*
 * To change this license header, choose License Headers in Project Properties.
```

```
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.ups.vista;

import ec.edu.ups.controlador.ControladorPrincipal;
import java.util.Calendar;
import javax.swing.table.DefaultTableModel;
import ec.edu.ups.modelo.Hipoteca;
import ec.edu.ups.modelo.Persona;
/**
 *
 * @author ASUS
 */
public class VentanaPagos extends javax.swing.JPanel {

    private ControladorPrincipal controladorPrincipal;
    private DefaultTableModel defaultTableModel;
    Persona persona;
    Hipoteca h;

    /**
     * Creates new form VentanaPagos
     */
    public VentanaPagos(ControladorPrincipal controladorPrincipal) {
        this.controladorPrincipal = controladorPrincipal;
//        defaultTableModel = (DefaultTableModel) tabla.getModel();
        initComponents();
    }
```

```
/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jScrollPane1 = new javax.swing.JScrollPane();
    tabla = new javax.swing.JTable();
    jLabel1 = new javax.swing.JLabel();
    txtCedula = new javax.swing.JTextField();
    buscar = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();
    jScrollPane2 = new javax.swing.JScrollPane();
    txtDatos = new javax.swing.JTextArea();

    tabla.setModel(new javax.swing.table.DefaultTableModel(
        new Object [][] {
            {null, null, null},
            {null, null, null},
            {null, null, null},
            {null, null, null}
        },
        new String [] {
            "Fechas", "n Pago", "Estado"
        }
    ));
```

```
jScrollPane1.setViewportView(tabla);

jLabel1.setText("Cedula ");

buscar.setText("Buscar");
buscar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        buscarActionPerformed(evt);
    }
});

jButton2.setText("Pagar");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

txtDatos.setColumns(20);
txtDatos.setRows(5);
jScrollPane2.setViewportView(txtDatos);

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(19, 19, 19)
```

```
                    .addComponent(jLabel1)
                    .addGap(18, 18, 18)
                    .addComponent(txtCedula, javax.swing.GroupLayout.PREFERRED_SIZE, 126, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(30, 30, 30)
                    .addComponent(buscar))
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                        .addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 262, javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addComponent(jScrollPane2, javax.swing.GroupLayout.Alignment.LEADING, javax.swing.GroupLayout.PREFERRED_SIZE, 275, javax.swing.GroupLayout.PREFERRED_SIZE))))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 94, Short.MAX_VALUE)
            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 434, javax.swing.GroupLayout.PREFERRED_SIZE))
        );
        jPanel1Layout.setVerticalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(29, 29, 29)
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(txtCedula, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(buscar)
                    .addComponent(jLabel1))
                .addGap(18, 18, 18)
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE)
```

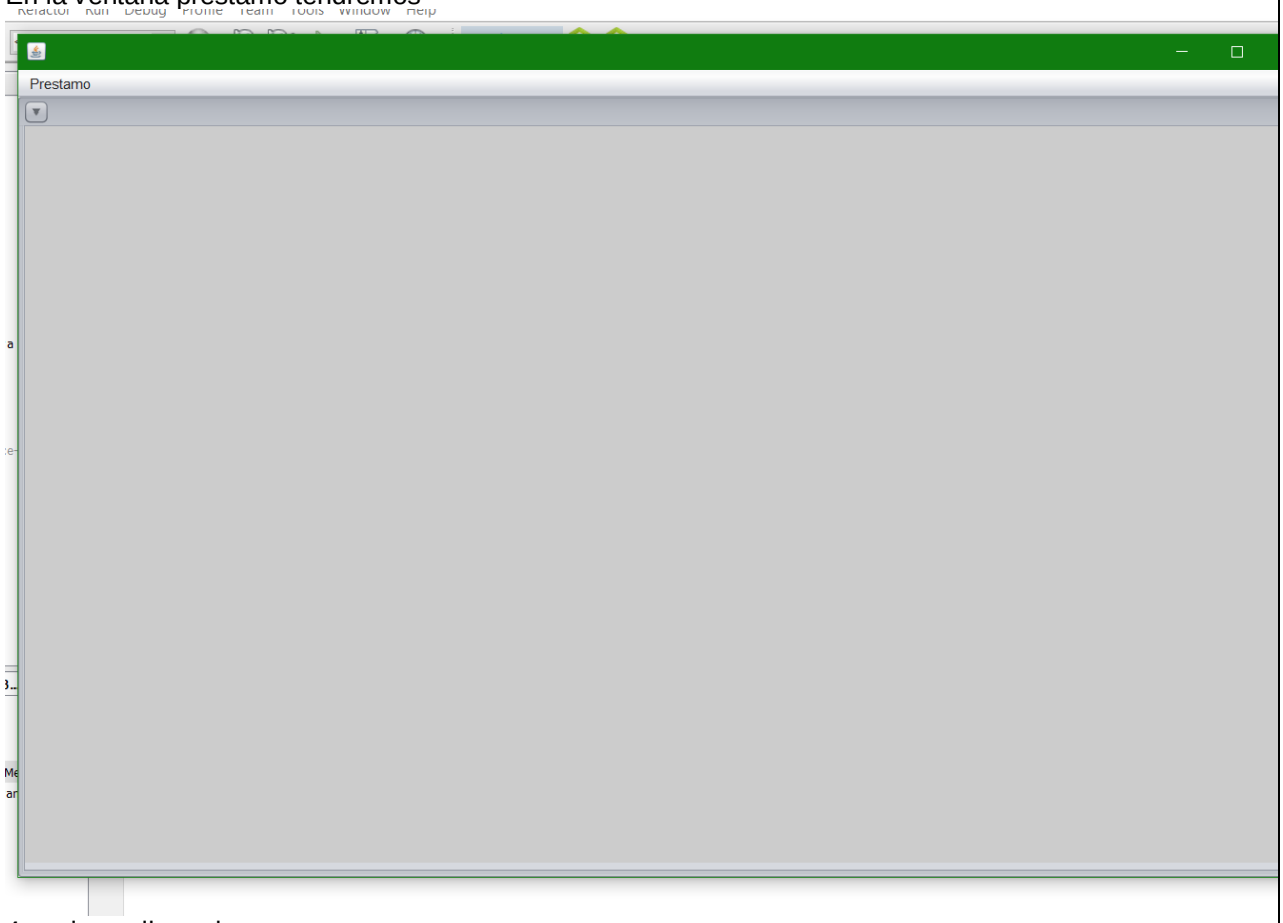
```
                .addComponent(jScrollPane2, javax.swing.GroupLayout.DEFAULT_SIZE, 251,
Short.MAX_VALUE))
                .addGap(18, 18, 18)
                .addComponent(jButton2)
                .addContainerGap(164, Short.MAX_VALUE))
        );

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(this);
        this.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)
        );
    }// </editor-fold>

    private void buscarActionPerformed(java.awt.event.ActionEvent evt) {
        persona = controladorPrincipal.BuscarPersona(txtCedula.getText());
        String d;

        for (Hipoteca hipoteca : controladorPrincipal.getHipotecaDAO().getListado()) {
            if (hipoteca.getDeudor().equals(persona)) {
                cargarDatosTabla(hipoteca);
                h = hipoteca;
                d = persona.toString() + h.toString();
                txtDatos.setText(d);
```

```
            } else {
            }
        }


    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        controladorPrincipal.getHipotecaDAO().actualizarPago(h);
        String d = persona.toString() + h.toString();
        txtDatos.setText(d);
        cargarDatosTabla(h);
    }

    public void cargarDatosTabla(Hipoteca hipoteca) {
        defaultTableModel = (DefaultTableModel) tabla.getModel();
        try {

            Calendar fecha = hipoteca.getFechaInicio();
            int numero = hipoteca.getNumeroPagos() - hipoteca.getPagosFaltantes();

            int n = hipoteca.getNumeroPagos();
            defaultTableModel.setRowCount(0);
            int contador = 0;
            String dia = Integer.toString(fecha.get(Calendar.DAY_OF_MONTH));
            String mes = Integer.toString(fecha.get(Calendar.MONTH));
            String año = Integer.toString(fecha.get(Calendar.YEAR));
            int m = Integer.valueOf(mes) + 1;
            mes = String.valueOf(m);
            for (int i = 0; i < n; i++) {
```

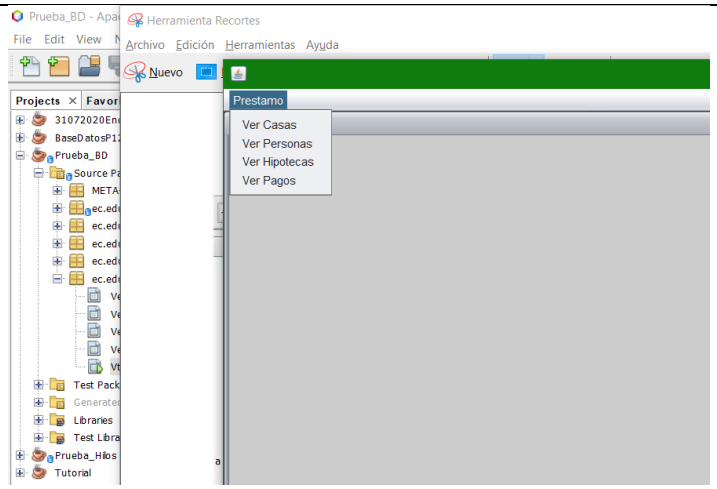
```
                contador += 1;
                String palabra;
                if (contador>numero) {
                    palabra= "Pendiente";
                }else{
                    palabra= "Pagado";
                }
                m = Integer.valueOf(mes) + 1;
                mes = String.valueOf(m);
                if (m > 12) {
                    int a = Integer.valueOf(año) + 1;
                    año = String.valueOf(a);
                    mes = "1";
                } else {
                }
                String fechaPago = dia + "/" + mes + "/" + año;
                String datos[] = {fechaPago, String.valueOf(contador),palabra};
                defaultTableModel.addRow(datos);
            }

        } catch (Exception e) {
        }
    }
    // Variables declaration - do not modify
    private javax.swing.JButton buscar;
    private javax.swing.JButton jButton2;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JScrollPane jScrollPane2;
```

```
    private javax.swing.JTable tabla;
    private javax.swing.JTextField txtCedula;
    private javax.swing.JTextArea txtDatos;
    // End of variables declaration
}
```

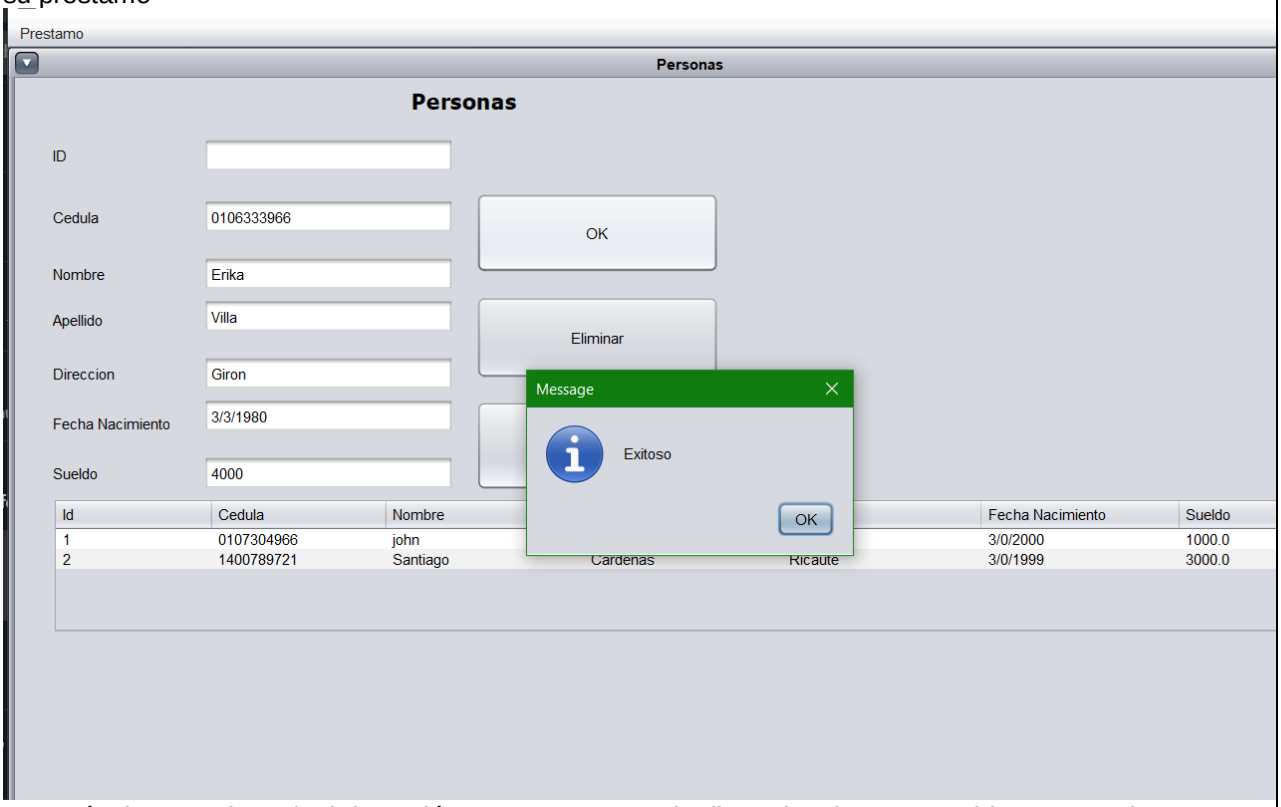
En la ejecución del programa tenemos que:
En la ventana préstamo tendremos



4 opciones llamadas

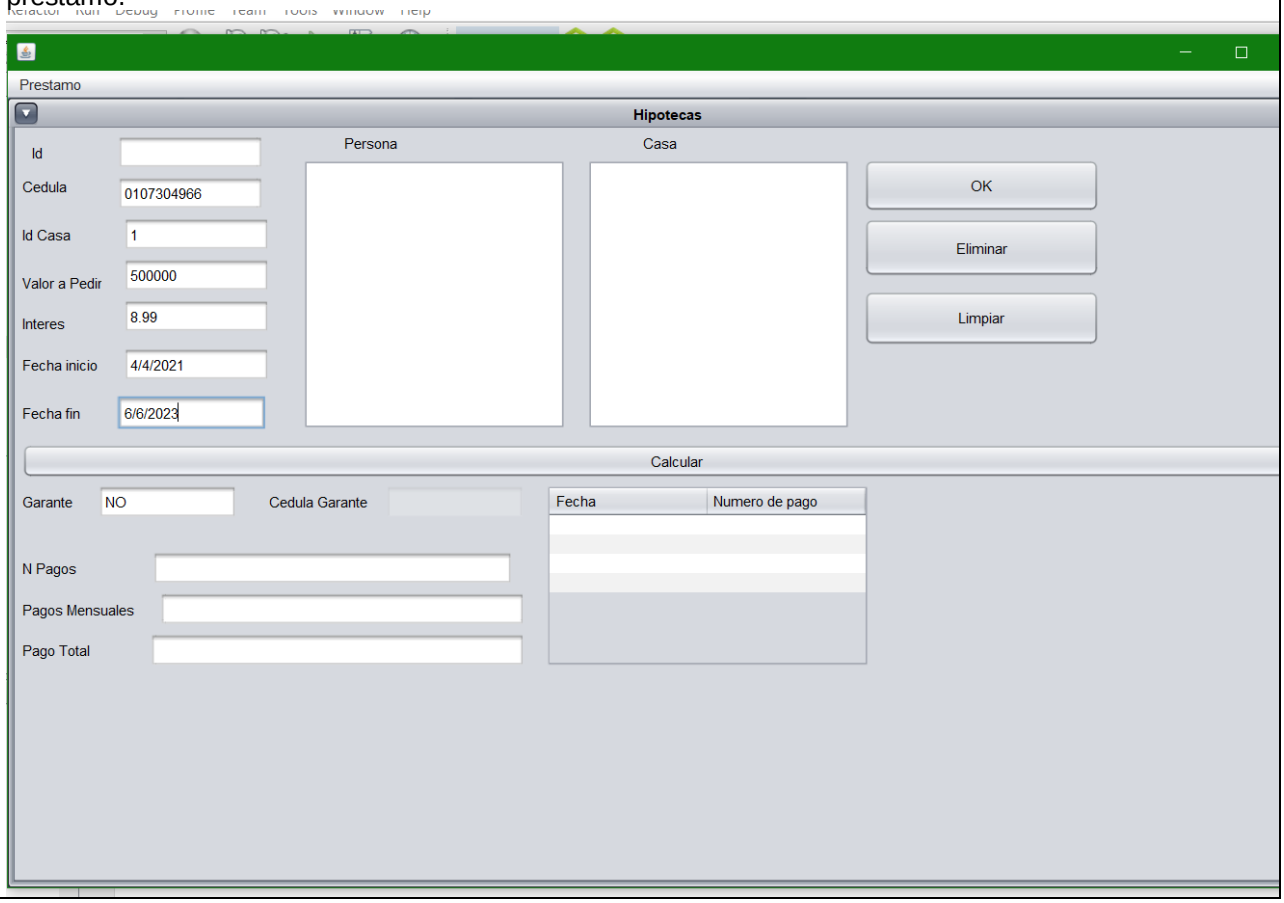


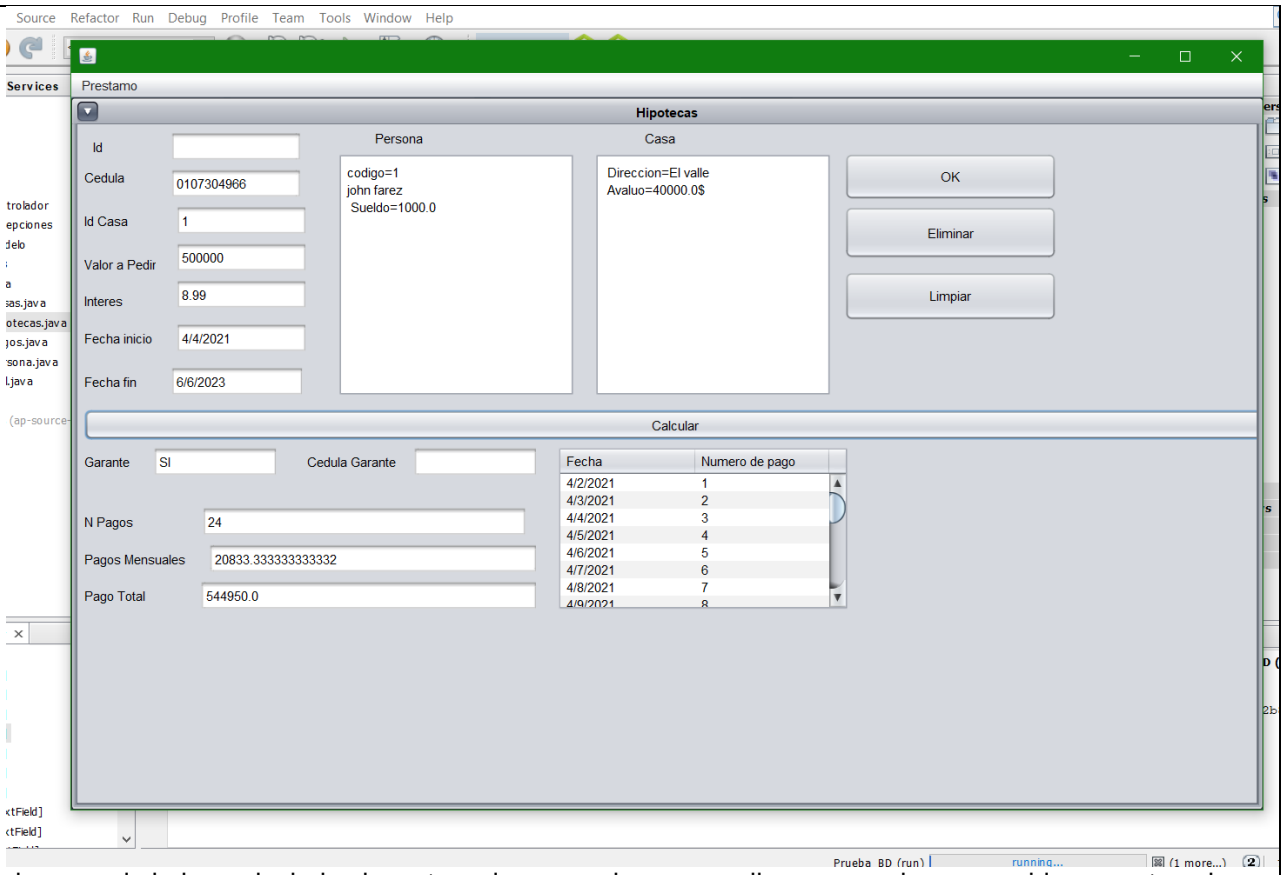
Ahora procederemos a ingresar en la opción de crear una persona para poder empezar con el tramite de su préstamo



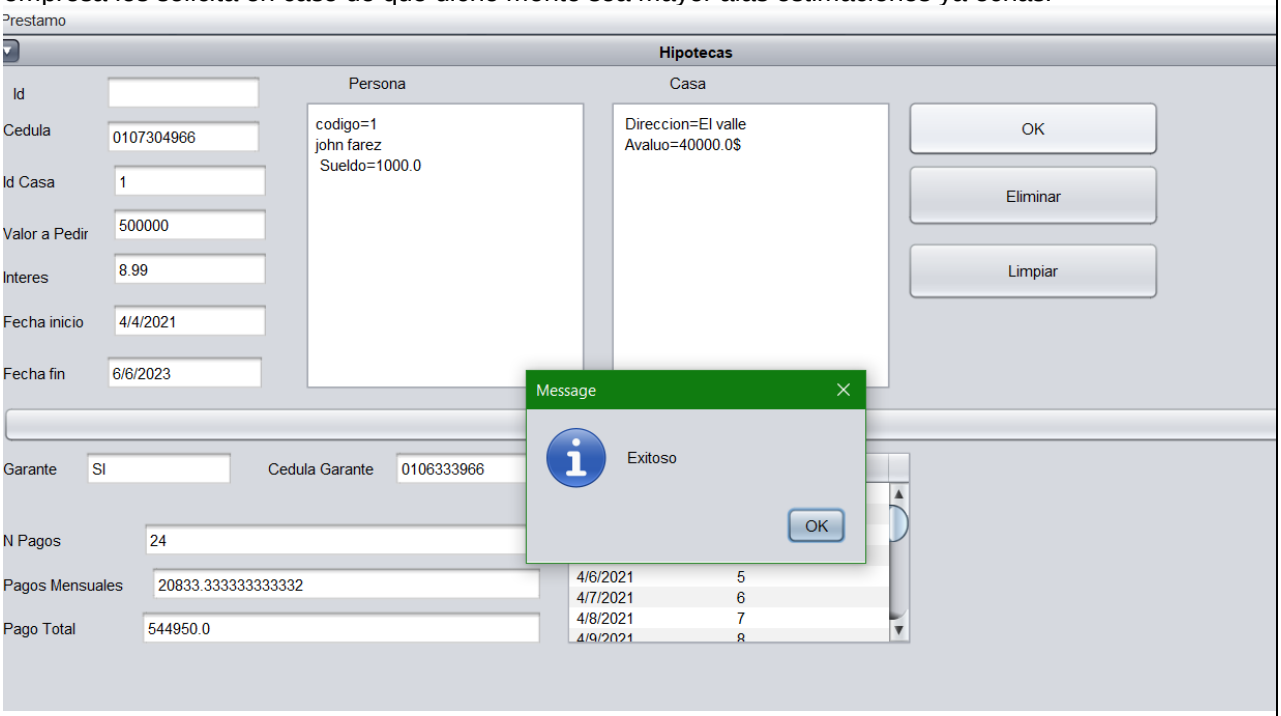
Después de procede a elegir la opción ver casa para poder llenar los datos requeridos para poder continuar con el procesos de el préstamo.

Después de procede con el ingrese de el ID de casa y un numero de cedula que será otorgada ono el prestamo.

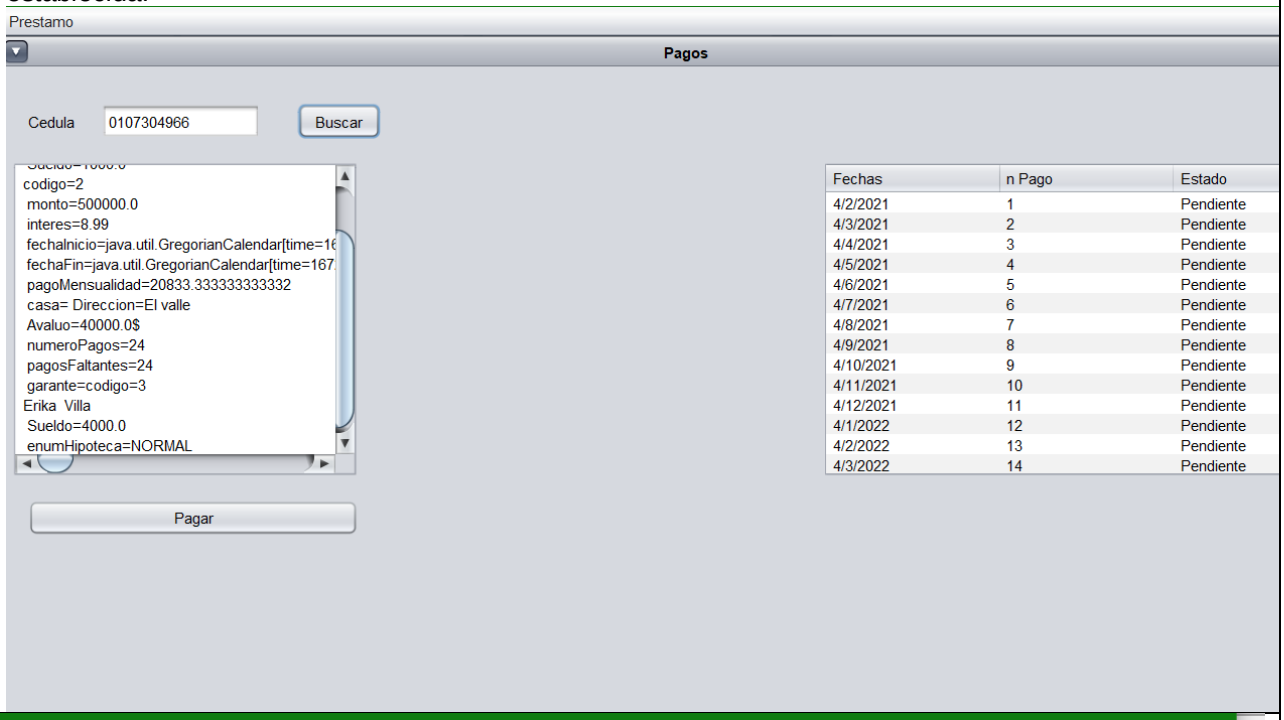


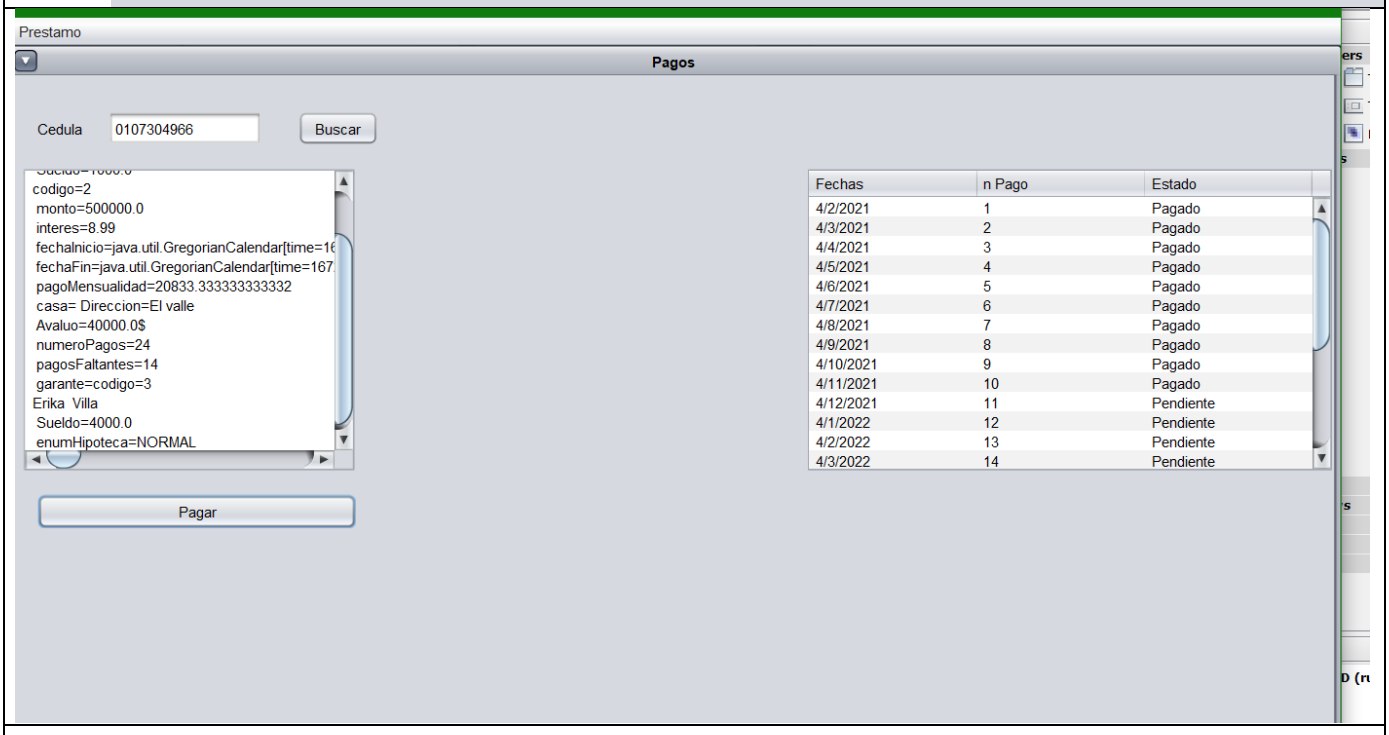


despues de haber calculado el monto y el numero de pagos a llevarse a cabo se considera oportuno la intervención de un garante para poder finalizar con el proceso del préstamo ya que las condiciones de la empresa los solicita en caso de que dicho monto sea mayor alas estimaciones ya echas.



Se controla el proceso de los pagos del deudor en este caso se observa como cambia el estado de pendiente a pagado pero por supuesto independientemente de cada numero de pago y la fecha establecida.





RESULTADO(S) OBTENIDO(S): Codificación de una manera oportuna todos los temas requerido en el sistema del bano además el uso correcto del JPA, MVC, GUI y el CRUD además de las excepciones.
CONCLUSIONES: Desarrollar buenas prácticas de programación aplicada dentro de un entorno o un régimen de codificion ya que las vistas en clase son las oportunas y las correctas.
RECOMENDACIONES: revisar mas la teoría para poder llevar a cabo soluciones mas rápidas a dichos conflictos que puedan aparecer en el desarrollo del programa

***Nombre de estudiante*: JohnFarez**

Firma de estudiante