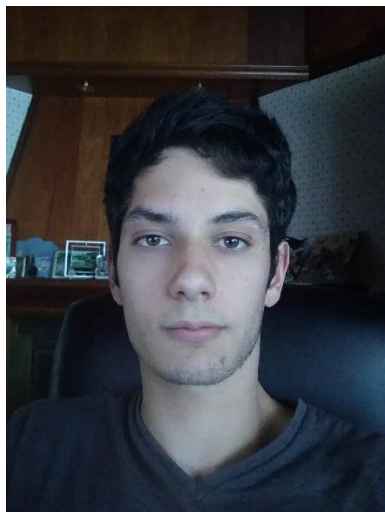


# Programación 2 Obligatorio 2

Ivan Kaitazoff

Numero de estudiante: 233940



Sebastian Romelli

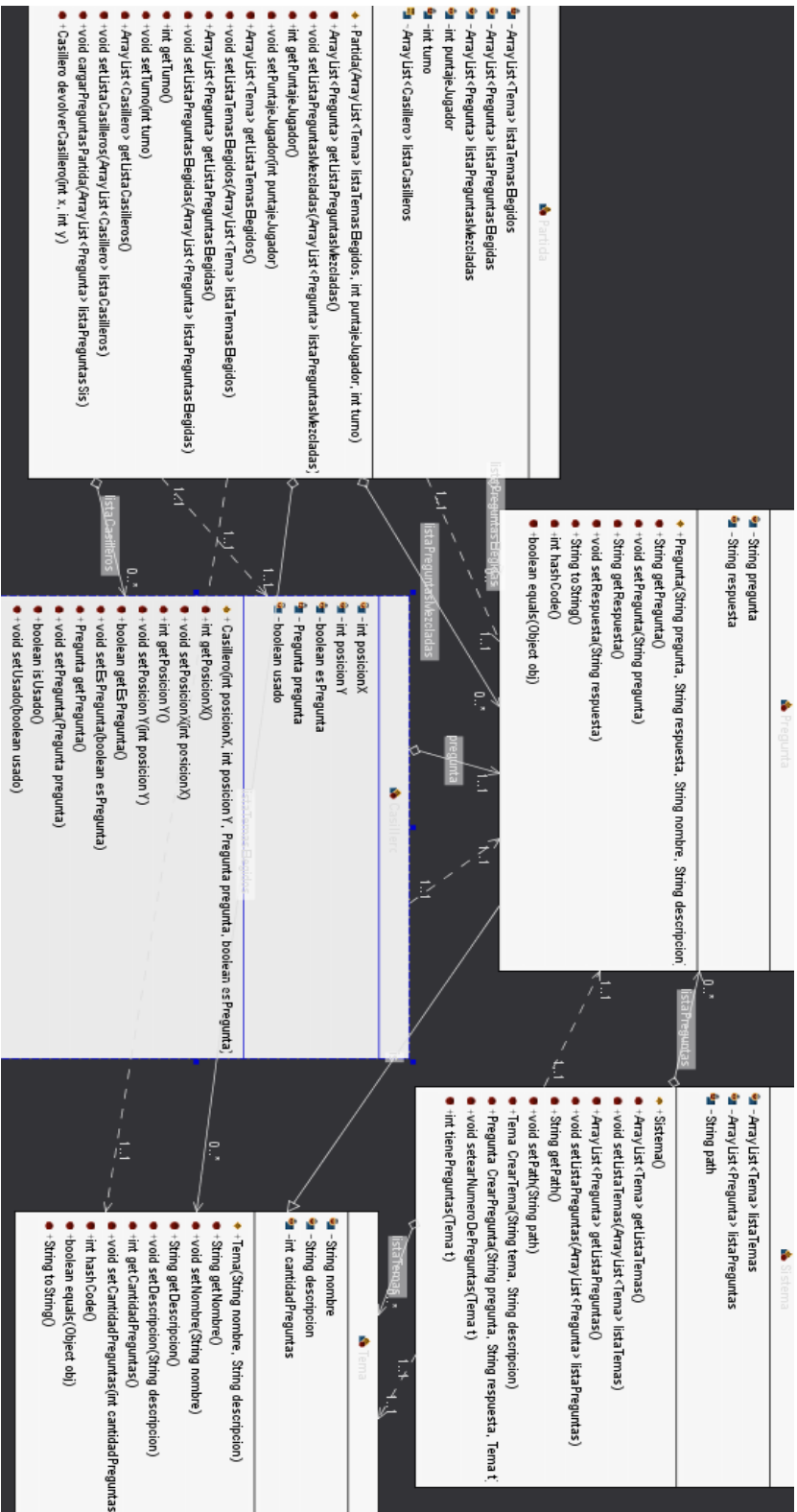
Numero de estudiante: 222405



# Link a video de presentación

[https://youtu.be/\\_qn757-5mR4](https://youtu.be/_qn757-5mR4)

# Diagrama UML



# Listado de código de fuente

## Clase Obligatorio2P2(main)

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package obligatorio2p2;

import dominio.Sistema;
import interfaz.MenuPrincipal;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;
import java.nio.file.Files;
import java.nio.file.Paths;
import javax.swing.JOptionPane;

public class Obligatorio2P2 {

    public static void main(String[] args) {
        Sistema sistema = new Sistema();
        sistema = cargarSistema();
        MenuPrincipal menu = new MenuPrincipal(sistema);

        menu.setVisible(true);
    }

    public static void guardarSistema(Sistema unSistema) {
        try {
            FileOutputStream fileOut = new FileOutputStream("datos.txt");
            ObjectOutputStream out = new ObjectOutputStream(fileOut);
            out.writeObject(unSistema);
            out.close();
        } catch (IOException e) {
            e.printStackTrace();
            JOptionPane.showMessageDialog(null, e.getMessage(), "Error",
JOptionPane.ERROR_MESSAGE);
        }
    }

    public static Sistema cargarSistema() {
        Sistema unSistema = new Sistema();
        try {
            ObjectInputStream in = new
ObjectInputStream(Files.newInputStream(Paths.get("datos.txt")));
            unSistema = (Sistema) in.readObject();
            in.close();
        } catch (IOException | ClassNotFoundException e) {
            JOptionPane.showMessageDialog(null, "No se encontro el archivo ",
"Error", JOptionPane.ERROR_MESSAGE);
        }
        return unSistema;
    }
}
```

```
}
```

# Clase ArchivoGrabacion

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package Persis;

import java.io.FileNotFoundException;
import java.util.Formatter;
import javax.swing.JOptionPane;

public class ArchivoGrabacion {

    private Formatter out;

    public ArchivoGrabacion(String unNombre) {
        try {
            out = new Formatter(unNombre);
        } catch (FileNotFoundException e) {
            JOptionPane.showMessageDialog(null, "No se pudo crear el
archivo", "Error", JOptionPane.ERROR_MESSAGE);
        } catch (SecurityException e) {
            JOptionPane.showMessageDialog(null, "No se tiene permiso",
"Error", JOptionPane.ERROR_MESSAGE);
        }
    }

    public void grabarLinea(String linea) {
        out.format(linea);
    }

    public void cerrar() {
        out.close();
    }
}
```

# Clase ArchivoLectura

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package Persis;

import java.io.*;
import java.nio.file.Paths;
import java.util.Scanner;
import javax.swing.JOptionPane;

public class ArchivoLectura {

    private Scanner in;
    private String linea;

    public ArchivoLectura(String unNombre) {
        try {
            in = new Scanner(Paths.get(unNombre));
        } catch (IOException e) {
            JOptionPane.showMessageDialog(null, "No se pudo leer el archivo",
"Error", JOptionPane.ERROR_MESSAGE);
        }
    }

    public boolean hayMasLineas() {
        boolean hay = false;
        linea = null;
        if (in.hasNext()) {
            linea = in.nextLine();
            hay = true;
        }
        return hay;
    }

    public String linea() {
        // devuelve la última línea leída
        return linea;
    }

    public void cerrar() {
        in.close();
    }
}
```

# Clase Casillero

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package dominio;

import java.awt.Button;
import javax.swing.JButton;

public class Casillero {

    private int posicionX;
    private int posicionY;
    private boolean esPregunta;
    private Pregunta pregunta;
    private boolean usado;

    public Casillero(int posicionX, int posicionY, Pregunta pregunta, boolean
esPregunta) {
        this.posicionX = posicionX;
        this.posicionY = posicionY;
        this.esPregunta = esPregunta;
        this.pregunta = pregunta;
        this.usado = false;
    }

    public int getPosicionX() {
        return posicionX;
    }

    public void setPosicionX(int posicionX) {
        this.posicionX = posicionX;
    }

    public int getPosicionY() {
        return posicionY;
    }

    public void setPosicionY(int posicionY) {
        this.posicionY = posicionY;
    }

    public boolean getEsPregunta() {
        return esPregunta;
    }

    public void setEsPregunta(boolean esPregunta) {
        this.esPregunta = esPregunta;
    }

    public Pregunta getPregunta() {
        return pregunta;
    }

    public void setPregunta(Pregunta pregunta) {
```

```
        this.pregunta = pregunta;
    }

    public boolean isUsado() {
        return usado;
    }

    public void setUsado(boolean usado) {
        this.usado = usado;
    }
}
```



# Clase Partida

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package dominio;

import java.util.ArrayList;

public class Partida {

    private ArrayList<Tema> listaTemasElegidos;
    private ArrayList<Pregunta> listaPreguntasElegidas;
    private ArrayList<Pregunta> listaPreguntasMezcladas;
    private int puntajeJugador;
    private int turno;
    ArrayList<Casillero> listaCasilleros;

    public Partida(ArrayList<Tema> listaTemasElegidos, int puntajeJugador,
int turno) {
        this.listaTemasElegidos = listaTemasElegidos;
        this.listaPreguntasElegidas = new ArrayList();
        this.puntajeJugador = puntajeJugador;
        this.turno = turno;
        this.listaCasilleros = new ArrayList<>();
    }

    public ArrayList<Pregunta> getListaPreguntasMezcladas() {
        return listaPreguntasMezcladas;
    }

    public void setListaPreguntasMezcladas(ArrayList<Pregunta>
listaPreguntasMezcladas) {
        this.listaPreguntasMezcladas = listaPreguntasMezcladas;
    }

    public int getPuntajeJugador() {
        return puntajeJugador;
    }

    public void setPuntajeJugador(int puntajeJugador) {
        this.puntajeJugador = puntajeJugador;
    }

    public ArrayList<Tema> getListaTemasElegidos() {
        return listaTemasElegidos;
    }

    public void setListaTemasElegidos(ArrayList<Tema> listaTemasElegidos) {
        this.listaTemasElegidos = listaTemasElegidos;
    }

    public ArrayList<Pregunta> getListaPreguntasElegidas() {
        return listaPreguntasElegidas;
    }
}
```

```

        public void setListaPreguntasElegidas(ArrayList<Pregunta>
listaPreguntasElegidas) {
            this.listaPreguntasElegidas = listaPreguntasElegidas;
        }

        public int getTurno() {
            return turno;
        }

        public void setTurno(int turno) {
            this.turno = turno;
        }

        public ArrayList<Casillero> getListaCasilleros() {
            return listaCasilleros;
        }

        public void setListaCasilleros(ArrayList<Casillero> listaCasilleros) {
            this.listaCasilleros = listaCasilleros;
        }

        public void cargarPreguntasPartida(ArrayList<Pregunta> listaPreguntasSis)
{
            if (!listaPreguntasSis.isEmpty() && !listaTemasElegidos.isEmpty()) {
                for (Tema t : listaTemasElegidos) {
                    for (Pregunta p : listaPreguntasSis) {
                        if (t.getNombre() == (p.getNombre())) {
                            listaPreguntasElegidas.add(p);
                        }
                    }
                }
            }
        }

        public Casillero devolverCasillero(int x, int y) {
            Casillero c = null;
            for (int i = 0; i < listaCasilleros.size(); i++) {
                if (listaCasilleros.get(i).getPosicionX() == x &&
listaCasilleros.get(i).getPosicionY() == y) {
                    c = listaCasilleros.get(i);
                }
            }
            return c;
        }
    }
}

```

# Clase Pregunta

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package dominio;

import java.util.Objects;

public class Pregunta extends Tema {

    private String pregunta;
    private String respuesta;

    public Pregunta(String pregunta, String respuesta, String nombre, String
descripcion) {
        super(nombre, descripcion);
        this.pregunta = pregunta;
        this.respuesta = respuesta;
    }

    public String getPregunta() {
        return pregunta;
    }

    public void setPregunta(String pregunta) {
        this.pregunta = pregunta;
    }

    public String getRespuesta() {
        return respuesta;
    }

    public void setRespuesta(String respuesta) {
        this.respuesta = respuesta;
    }

    @Override
    public String toString() {
        return pregunta + ", respuesta=" + respuesta + ", tema=" +
this.getNombre();
    }

    @Override
    public int hashCode() {
        int hash = 7;
        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 Pregunta other = (Pregunta) obj;  
    if (!Objects.equals(this.pregunta, other.pregunta)) {  
        return false;  
    }  
    return true;  
}  
}
```

# Clase Sistema

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package dominio;

import java.io.Serializable;
import java.util.ArrayList;
import javax.swing.JOptionPane;

public class Sistema implements Serializable {

    private ArrayList<Tema> listaTemas;
    private ArrayList<Pregunta> listaPreguntas;
    private String path;

    public Sistema() {
        this.listaTemas = new ArrayList<>();
        this.listaPreguntas = new ArrayList<>();
    }

    public ArrayList<Tema> getListasTemas() {
        return listaTemas;
    }

    public void setListasTemas(ArrayList<Tema> listaTemas) {
        this.listaTemas = listaTemas;
    }

    public ArrayList<Pregunta> getListasPreguntas() {
        return listaPreguntas;
    }

    public void setListasPreguntas(ArrayList<Pregunta> listaPreguntas) {
        this.listaPreguntas = listaPreguntas;
    }

    public String getPath() {
        return path;
    }

    public void setPath(String path) {
        this.path = path;
    }

    public Tema CrearTema(String tema, String descripcion) {
        Tema t = new Tema(tema, descripcion);
        listaTemas.add(t);
        return t;
    }

    public Pregunta CrearPregunta(String pregunta, String respuesta, Tema t)
    {
        //Preguntas(String pregunta, String respuesta, String nombre, String
        descripcion)
    }
}
```

```

        Pregunta p = new Pregunta(pregunta, respuesta, t.getNombre(),
t.getDescripcion());
        listaPreguntas.add(p);
        return p;
    }

    public void setearNumeroDePreguntas(Tema t) {
        int indexTema = 0;
        for (int i = 0; i < listaTemas.size(); i++) {
            if (t.getNombre().equals(listaTemas.get(i).getNombre())) {
                indexTema = i;
                break;
            }
        }
        int preguntas = tienePreguntas(t);
        listaPreguntas.get(indexTema).setCantidadPreguntas(preguntas);
    }

    public int tienePreguntas(Tema t) {
        int contador = 0;
        try {
            for (Pregunta p : listaPreguntas) {
                if (p.getNombre().equals(t.getNombre())) {
                    contador++;
                }
            }
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null, "Error leyendo preguntas",
"Error", JOptionPane.ERROR_MESSAGE);
        }

        return contador;
    }
}

```

# Clase Tema

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package dominio;

import java.io.Serializable;
import java.util.ArrayList;
import java.util.Objects;

public class Tema implements Serializable {

    private String nombre;
    private String descripcion;
    private int cantidadPreguntas;

    public Tema(String nombre, String descripcion) {
        this.nombre = nombre;
        this.descripcion = descripcion;
    }

    public String getNombre() {
        return nombre;
    }

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

    public String getDescripcion() {
        return descripcion;
    }

    public void setDescripcion(String descripcion) {
        this.descripcion = descripcion;
    }

    public int getCantidadPreguntas() {
        return cantidadPreguntas;
    }

    public void setCantidadPreguntas(int cantidadPreguntas) {
        this.cantidadPreguntas = cantidadPreguntas;
    }

    @Override
    public int hashCode() {
        int hash = 3;
        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 Tema other = (Tema) obj;
    if (!Objects.equals(this.nombre, other.nombre)) {
        return false;
    }
    return true;
}

@Override
public String toString() {
    return nombre + "," + cantidadPreguntas + "(cantidad preguntas)";
}

}

```



# (JFrame) BuscarDocumento

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Sistema;
import java.awt.event.WindowEvent;
import javax.swing.JFileChooser;
import javax.swing.JOptionPane;

public class BuscarDocumento extends javax.swing.JFrame {

    private String path;
    private Sistema sistema;

    public BuscarDocumento(Sistema sistema) {
        initComponents();
        this.sistema = sistema;
        this.setLocationRelativeTo(null);
    }

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

        btnSalir = new javax.swing.JToggleButton();
        jFileChooser1 = new javax.swing.JFileChooser();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setMaximumSize(new java.awt.Dimension(600, 600));
        setMinimumSize(new java.awt.Dimension(600, 400));
        setUndecorated(true);
        setPreferredSize(new java.awt.Dimension(730, 450));
        getContentPane().setLayout(null);

        btnSalir.setText("Salir");
        btnSalir.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnSalirActionPerformed(evt);
            }
        });
        getContentPane().add(btnSalir);
        btnSalir.setBounds(610, 360, 80, 30);

        jFileChooser1.setBackground(new java.awt.Color(187, 83, 70));
        jFileChooser1.setDialogTitle("");
        jFileChooser1.setFileFilter(null);
        jFileChooser1.setForeground(new java.awt.Color(187, 83, 70));
        jFileChooser1.setMinimumSize(new java.awt.Dimension(600, 400));
        jFileChooser1.setPreferredSize(new java.awt.Dimension(600, 400));
        jFileChooser1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jFileChooser1ActionPerformed(evt);
            }
        });
    }
}
```

```

        }
    });
    getContentPane().add(jFileChooser1);
    jFileChooser1.setBounds(0, 0, 700, 400);

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

private void jFileChooser1ActionPerformed(java.awt.event.ActionEvent evt)
{
    try {
        this.setPath(jFileChooser1.getSelectedFile().toString());
        System.out.println(path);
        sistema.setPath(this.getPath());

    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, "Error cargando el path",
"Error", JOptionPane.ERROR_MESSAGE);
    }

}

private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose();
}

public String getPath() {
    return path;
}

public void setPath(String path) {
    this.path = path;
}

public JFileChooser getjFileChooser1() {
    return jFileChooser1;
}

public void setjFileChooser1(JFileChooser jFileChooser1) {
    this.jFileChooser1 = jFileChooser1;
}

public void windowClosing(WindowEvent e) {
    System.out.println("cerro la ventana");
}

// Variables declaration - do not modify
private javax.swing.JToggleButton btnSalir;
private javax.swing.JFileChooser jFileChooser1;
// End of variables declaration
}

```

# (JFrame) CargaDatosPrueba

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Pregunta;
import dominio.Sistema;
import dominio.Tema;

public class CargaDatosPrueba extends javax.swing.JFrame {

    private Sistema sistema;
    private int cantPreguntas;
    private int cantTemas;

    public CargaDatosPrueba(Sistema sistema) {
        initComponents();
        this.setLocationRelativeTo(null);
        this.sistema = sistema;
    }

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

        jPanel1 = new javax.swing.JPanel();
        jTextCantTemas = new javax.swing.JTextField();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jTextCantPreguntas = new javax.swing.JTextField();
        btnCrearDatos = new javax.swing.JButton();
        btnSalir = new javax.swing.JButton();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setUndecorated(true);
        setPreferredSize(new java.awt.Dimension(450, 300));
        getContentPane().setLayout(null);

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

        jTextCantTemas.setBackground(new java.awt.Color(187, 83, 70));

        jTextCantTemas.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
        jPanel1.add(jTextCantTemas);
        jTextCantTemas.setBounds(215, 35, 167, 18);

        jLabel1.setText("Cantidad de temas");
        jPanel1.add(jLabel1);
```

```

jLabel1.setBounds(43, 38, 90, 14);

jLabel2.setText("Cantidad de preguntas en cada tema");
jPanel1.add(jLabel2);
jLabel2.setBounds(0, 82, 190, 14);

jTextCantPreguntas.setBackground(new java.awt.Color(187, 83, 70));

jTextCantPreguntas.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
jPanel1.add(jTextCantPreguntas);
jTextCantPreguntas.setBounds(215, 79, 167, 18);

btnCrearDatos.setBackground(new java.awt.Color(187, 83, 70));
btnCrearDatos.setText("Crear Datos");

btnCrearDatos.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
btnCrearDatos.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnCrearDatosActionPerformed(evt);
    }
});
jPanel1.add(btnCrearDatos);
btnCrearDatos.setBounds(230, 140, 130, 19);

btnSalir.setBackground(new java.awt.Color(187, 83, 70));
btnSalir.setText("Salir");

btnSalir.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
btnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSalirActionPerformed(evt);
    }
});
jPanel1.add(btnSalir);
btnSalir.setBounds(40, 140, 120, 19);

jLabel3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel3.setText("AVISO, AL CREAR DATOS USANDO ESTA VENTANA SE BORRARAN");
jPanel1.add(jLabel3);
jLabel3.setBounds(0, 190, 400, 20);

jLabel4.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel4.setText(" TODOS LOS DATOS QUE HAY EN EL SISTEMA");
jPanel1.add(jLabel4);
jLabel4.setBounds(0, 220, 400, 14);

getContentPane().add(jPanel1);
jPanel1.setBounds(0, 0, 400, 270);

pack();

```

```

} // </editor-fold>

private void btnCrearDatosActionPerformed(java.awt.event.ActionEvent evt)
{
    sistema.getListaPreguntas().clear();
    sistema.getListaTemas().clear();
    cantTemas = Integer.parseInt(jTextCantTemas.getText());
    cantPreguntas = Integer.parseInt(jTextCantPreguntas.getText());
    for (int i = 0; i < cantTemas; i++) {
        Tema t = new Tema("TEMA" + (i + 1), ", Descripcion de T" + (i +
1));
        sistema.getListaTemas().add(t);
        for (int j = 0; j < cantPreguntas; j++) {
            Pregunta p = new Pregunta("P" + (j + 1) + "TEMA" + (i + 1),
"Respuesta P" + (j + 1) + "TEMA" + (i + 1), t.getNombre(),
t.getDescripcion());
            sistema.getListaPreguntas().add(p);
            t.setCantidadPreguntas(t.getCantidadPreguntas() + 1);
        }
    }
    this.dispose();
}

private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose();
}

// Variables declaration - do not modify
private javax.swing.JButton btnCrearDatos;
private javax.swing.JButton btnSalir;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JPanel jPanel1;
private javax.swing.JTextField jTextCantPreguntas;
private javax.swing.JTextField jTextCantTemas;
// End of variables declaration
}

```

# (JFrame) CargaDatosVentanaAuxiliar

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import Persis.ArchivoGrabacion;
import Persis.ArchivoLectura;
import dominio.Pregunta;
import dominio.Sistema;
import java.util.ArrayList;
import javax.swing.JOptionPane;

public class CargaDatosVentanaAuxiliar extends javax.swing.JFrame {

    Sistema sistema;

    public CargaDatosVentanaAuxiliar(Sistema sistema) {
        initComponents();
        this.sistema = sistema;
        this.setLocationRelativeTo(null);
    }

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

        panell = new java.awt.Panel();
        cargaDatos = new javax.swing.JButton();
        mostrarRuta = new javax.swing.JButton();
        abrirNavegador = new javax.swing.JButton();
        labelPath = new javax.swing.JLabel();
        btnSalir = new javax.swing.JButton();
        cargaPrueba = new javax.swing.JButton();
        guardarDatos = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setBackground(new java.awt.Color(255, 255, 255));
        setUndecorated(true);
        setPreferredSize(new java.awt.Dimension(600, 400));
        getContentPane().setLayout(null);

        panell.setLayout(null);

        cargaDatos.setBackground(new java.awt.Color(187, 83, 70));
        cargaDatos.setText("Cargar datos desde txt");

        cargaDatos.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.
border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
        cargaDatos.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                cargaDatosActionPerformed(evt);
            }
        })
    }
}
```

```

});
panell1.add(cargaDatos);
cargaDatos.setBounds(30, 170, 150, 19);

mostrarRuta.setBackground(new java.awt.Color(187, 83, 70));
mostrarRuta.setText("Mostrar ruta seleccionada");

mostrarRuta.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
mostrarRuta.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        mostrarRutaActionPerformed(evt);
    }
});
panell1.add(mostrarRuta);
mostrarRuta.setBounds(70, 70, 250, 19);

abrirNavegador.setBackground(new java.awt.Color(187, 83, 70));
abrirNavegador.setText("Abrir navegador");

abrirNavegador.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.sw
ing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
abrirNavegador.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        abrirNavegadorActionPerformed(evt);
    }
});
panell1.add(abrirNavegador);
abrirNavegador.setBounds(70, 40, 250, 19);

labelPath.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
labelPath.setText("Path");
panell1.add(labelPath);
labelPath.setBounds(10, 10, 380, 14);

btnSalir.setBackground(new java.awt.Color(187, 83, 70));
btnSalir.setText("Salir");

btnSalir.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
btnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSalirActionPerformed(evt);
    }
});
panell1.add(btnSalir);
btnSalir.setBounds(150, 260, 100, 19);

cargaPrueba.setBackground(new java.awt.Color(187, 83, 70));
cargaPrueba.setText("Carga facil de datos de prueba");

cargaPrueba.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing

```

```

.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
    cargaPrueba.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            cargaPruebaActionPerformed(evt);
        }
    });
    panel1.add(cargaPrueba);
    cargaPrueba.setBounds(60, 220, 290, 19);

    guardarDatos.setBackground(new java.awt.Color(187, 83, 70));
    guardarDatos.setText("Guardar datos a txt");

    guardarDatos.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swin
g.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
    guardarDatos.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            guardarDatosActionPerformed(evt);
        }
    });
    panel1.add(guardarDatos);
    guardarDatos.setBounds(220, 170, 170, 19);

    getContentPane().add(panel1);
    panel1.setBounds(0, 0, 400, 300);

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

private void abrirNavegadorActionPerformed(java.awt.event.ActionEvent
evt) {
    BuscarDocumento b = new BuscarDocumento(sistema);
    b.setVisible(true);
}

private void mostrarRutaActionPerformed(java.awt.event.ActionEvent evt) {
    labelPath.setText(sistema.getPath());
}

private void cargaDatosActionPerformed(java.awt.event.ActionEvent evt) {
    try{ArchivoLectura arch = new ArchivoLectura(sistema.getPath());
    ArrayList<String> listaCargados = new ArrayList<>();
    while (arch.hayMasLineas()) {
        listaCargados.add(arch.linea());
    }
    arch.cerrar();
    boolean sePuedeAgregar = false;
    for (int i = 0; i < listaCargados.size(); i += 3) {
        if (listaCargados.get(i) == "" || listaCargados.get(i + 1) == ""
|| listaCargados.get(i + 2) == "") {
            JOptionPane.showMessageDialog(null, "Hay espacios en blanco,
no se pudo agregar la pregunta", "Error", JOptionPane.ERROR_MESSAGE);
        } else {
            for (int j = 0; j < sistema.getListaTemas().size(); j++) {
                if (listaCargados.get(i + 1) != "" && listaCargados.get(i
+ 2) != "") {

```



```

        if
(sistema.getListasTemas().get(j).getNombre().equals(listaCargados.get(i).toUpperCase())) {
            Pregunta p = new Pregunta(listaCargados.get(i +
1).toString(), listaCargados.get(i + 2).toString(),

sistema.getListasTemas().get(j).getNombre(),
sistema.getListasTemas().get(j).getDescripcion());
            sistema.getListasPreguntas().add(p);

sistema.getListasTemas().get(j).setCantidadPreguntas(sistema.getListasTemas().get(j).getCantidadPreguntas() + 1);
            System.out.println(p.toString());
        }
    }

}

}}
catch(Exception e){
    JOptionPane.showMessageDialog(null, "No se pudo encontrar el
archivo", "Error", JOptionPane.ERROR_MESSAGE);
}

}

private void guardarDatosActionPerformed(java.awt.event.ActionEvent evt)
{
    try{
        ArchivoGrabacion out = new ArchivoGrabacion(sistema.getPath());
        for (int i = 0; i < sistema.getListasTemas().size(); i++) {
            out.grabarLinea("Tema
" + (i + 1) + ": " + sistema.getListasTemas().get(i).getNombre() + "\n");
            for (int j = 0; j < sistema.getListasPreguntas().size(); j++)
            {
                if
(sistema.getListasPreguntas().get(j).getNombre().equals(sistema.getListasTemas().get(i).getNombre())) {
                    out.grabarLinea("Pregunta: " +
sistema.getListasPreguntas().get(j).getPregunta() + "\n");
                    out.grabarLinea("Respuesta: " +
sistema.getListasPreguntas().get(j).getRespuesta() + "\n");
                }
            }
        }

        out.cerrar();
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, "No se pudo encontrar el
archivo", "Error", JOptionPane.ERROR_MESSAGE);
    }
}

private void cargaPruebaActionPerformed(java.awt.event.ActionEvent evt) {
    CargaDatosPrueba c = new CargaDatosPrueba(sistema);
    c.setVisible(true);
}

```

```
}

private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose();
}

// Variables declaration - do not modify
private javax.swing.JButton abrirNavegador;
private javax.swing.JButton btnSalir;
private javax.swing.JButton cargaDatos;
private javax.swing.JButton cargaPrueba;
private javax.swing.JButton guardarDatos;
private javax.swing.JLabel labelPath;
private javax.swing.JButton mostrarRuta;
private java.awt.Panel panel1;
// End of variables declaration
}
```

# (JFrame) GestionPregunta

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Pregunta;
import dominio.Sistema;
import dominio.Tema;

public class GestionPregunta extends javax.swing.JFrame {

    Sistema sistema;

    public GestionPregunta(Sistema sistema) {
        this.sistema = sistema;
        initComponents();
        cargarTemas();
        jListaPreguntas.setListData(sistema.getListaPreguntas().toArray());
    }

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

        jPanel1 = new javax.swing.JPanel();
        cmbTemas = new javax.swing.JComboBox();
        volverAMenu = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        pregunta = new javax.swing.JTextField();
        respuesta = new javax.swing.JTextField();
        guardarPregunta = new javax.swing.JButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        jListaPreguntas = new javax.swing.JList();
        jButton1 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setUndecorated(true);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));
        jPanel1.setPreferredSize(new java.awt.Dimension(600, 400));
        jPanel1.setLayout(null);

        cmbTemas.setBackground(new java.awt.Color(187, 83, 70));
        cmbTemas.setToolTipText("");

        cmbTemas.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED));
        jPanel1.add(cmbTemas);
        cmbTemas.setBounds(140, 10, 254, 28);

        volverAMenu.setBackground(new java.awt.Color(187, 83, 70));
```

```

        volverAMenu.setForeground(new java.awt.Color(0, 0, 0));
        volverAMenu.setText("Volver a menu principal");

volverAMenu.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
        volverAMenu.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                volverAMenuActionPerformed(evt);
            }
        });
        jPanel1.add(volverAMenu);
        volverAMenu.setBounds(440, 310, 140, 19);

        jLabel1.setBackground(new java.awt.Color(204, 0, 204));
        jLabel1.setForeground(new java.awt.Color(0, 0, 0));
        jLabel1.setText("Seleccionar Tema");
        jPanel1.add(jLabel1);
        jLabel1.setBounds(20, 10, 110, 30);

        jLabel2.setBackground(new java.awt.Color(187, 83, 70));
        jLabel2.setForeground(new java.awt.Color(0, 0, 0));
        jLabel2.setText("Ingresar pregunta");
        jPanel1.add(jLabel2);
        jLabel2.setBounds(20, 80, 110, 40);

        jLabel3.setForeground(new java.awt.Color(0, 0, 0));
        jLabel3.setText("Ingresar respuesta");
        jPanel1.add(jLabel3);
        jLabel3.setBounds(20, 190, 92, 15);

        pregunta.setBackground(new java.awt.Color(187, 83, 70));

pregunta.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
        jPanel1.add(pregunta);
        pregunta.setBounds(140, 90, 254, 19);

        respuesta.setBackground(new java.awt.Color(187, 83, 70));

respuesta.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.b
order.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
        jPanel1.add(respuesta);
        respuesta.setBounds(140, 190, 254, 19);

        guardarPregunta.setBackground(new java.awt.Color(187, 83, 70));
        guardarPregunta.setForeground(new java.awt.Color(0, 0, 0));
        guardarPregunta.setText("Guardar Pregunta");

guardarPregunta.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.s
wing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
        guardarPregunta.addActionListener(new java.awt.event.ActionListener()
{
            public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        guardarPreguntaActionPerformed(evt);
    }
});
jPanell1.add(guardarPregunta);
guardarPregunta.setBounds(20, 310, 140, 19);

jListaPreguntas.setBackground(new java.awt.Color(187, 83, 70));

jListaPreguntas.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.s
wing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jScrollPane1.setViewportView(jListaPreguntas);

jPanell1.add(jScrollPane1);
jScrollPane1.setBounds(460, 20, 110, 200);

jButton1.setBackground(new java.awt.Color(187, 83, 70));
jButton1.setForeground(new java.awt.Color(0, 0, 0));
jButton1.setText("Eliminar tema seleccionado");

jButton1.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
jPanell1.add(jButton1);
jButton1.setBounds(410, 240, 170, 19);

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

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanell1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanell1, javax.swing.GroupLayout.DEFAULT_SIZE, 391,
Short.MAX_VALUE)
);

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

private void volverAMenuActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose();
}

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

```

```

        String textoPregunta = pregunta.getText();
        String textoRespuesta = respuesta.getText();
        Tema t = (Tema) cmbTemas.getSelectedItem();
        boolean repetido = false;
        if (!pregunta.getText().equals("") &&
!respuesta.getText().equals("")) {
            for (int i = 0; i < sistema.getListaPreguntas().size(); i++) {
                if
(sistema.getListaPreguntas().get(i).getPregunta().equals(textoPregunta) &&
sistema.getListaPreguntas().get(i).getNombre().equals(t.getNombre())) {

sistema.getListaPreguntas().get(i).setRespuesta(textoRespuesta);
                repetido = true;
            }
        }
        if (!repetido) {
            Pregunta p = new Pregunta(textoPregunta, textoRespuesta,
t.getNombre(), t.getDescripcion());
            sistema.getListaPreguntas().add(p);
            t.setCantidadPreguntas(t.getCantidadPreguntas() + 1);
        }
    }

    jListaPreguntas.setListData(sistema.getListaPreguntas().toArray());
    pregunta.setText("");
    respuesta.setText("");
    cargarTemas();
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    Tema t = sistema.getListaTemas().get(0);
    for (int i = 0; i < sistema.getListaTemas().size(); i++) {
        if
(sistema.getListaTemas().get(i).getNombre().equals(sistema.getListaPreguntas(
).get(jListaPreguntas.getSelectedIndex()).getNombre())) {
            t = sistema.getListaTemas().get(i);
        }
    }
    t.setCantidadPreguntas(t.getCantidadPreguntas() - 1);

sistema.getListaPreguntas().remove(jListaPreguntas.getSelectedIndex());
    jListaPreguntas.setListData(sistema.getListaPreguntas().toArray());
    cargarTemas();
}

// Variables declaration - do not modify
private javax.swing.JComboBox cmbTemas;
private javax.swing.JButton guardarPregunta;
private javax.swing.JButton jButton1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JList jListaPreguntas;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTextField pregunta;

```

```
private javax.swing.JTextField respuesta;
private javax.swing.JButton volverAMenu;
// End of variables declaration

private void cargarTemas() {
    cmbTemas.removeAllItems();
    if (!sistema.getListaTemas().isEmpty()) {
        for (Tema tema : sistema.getListaTemas()) {
            cmbTemas.addItem(tema);
        }
    }
}
}
```

# (JFrame) GestionTema

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Sistema;
import dominio.Tema;
import java.util.ArrayList;
import javax.swing.JOptionPane;

public class GestionTema extends javax.swing.JFrame {

    Sistema sistema;
    ArrayList<Tema> lista = new ArrayList();

    public GestionTema(ArrayList<Tema> listaTemas, Sistema sistema) {
        this.sistema = sistema;
        initComponents();
        this.setLocationRelativeTo(null);
        lista = listaTemas;
        setJListTema();
    }

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

        jPanel1 = new javax.swing.JPanel();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        VolverAMenu = new javax.swing.JButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        jListTema = new javax.swing.JList();
        botonEliminarTema = new javax.swing.JButton();
        jLabel3 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setUndecorated(true);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));
        jPanel1.setPreferredSize(new java.awt.Dimension(600, 400));
        jPanel1.setLayout(null);

        jTextField1.setBackground(new java.awt.Color(187, 83, 70));

        jTextField1.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
        .border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
        java.awt.Color.black, java.awt.Color.black));
        jTextField1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {

```



```

        jTextField1ActionPerformed(evt);
    }
});
jPanel1.add(jTextField1);
jTextField1.setBounds(190, 10, 188, 19);

jTextField2.setBackground(new java.awt.Color(187, 83, 70));

jTextField2.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jTextField2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField2ActionPerformed(evt);
    }
});
jPanel1.add(jTextField2);
jTextField2.setBounds(190, 90, 188, 60);

jLabel1.setBackground(new java.awt.Color(187, 83, 70));
jLabel1.setForeground(new java.awt.Color(0, 0, 0));
jLabel1.setText("Nombre del tema");
jPanel1.add(jLabel1);
jLabel1.setBounds(12, 14, 120, 15);

jLabel2.setBackground(new java.awt.Color(187, 83, 70));
jLabel2.setForeground(new java.awt.Color(0, 0, 0));
jLabel2.setText("Descripcion");
jPanel1.add(jLabel2);
jLabel2.setBounds(12, 112, 90, 15);

jButton1.setBackground(new java.awt.Color(187, 83, 70));
jButton1.setForeground(new java.awt.Color(0, 0, 0));
jButton1.setText("Guardar tema");

jButton1.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
jPanel1.add(jButton1);
jButton1.setBounds(12, 287, 130, 19);

VolverAMenu.setBackground(new java.awt.Color(187, 83, 70));
VolverAMenu.setForeground(new java.awt.Color(0, 0, 0));
VolverAMenu.setText("Volver a menu principal");

VolverAMenu.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
VolverAMenu.setCursor(new
java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
VolverAMenu.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        VolverAMenuActionPerformed(evt);
    }
});
jPanel1.add(VolverAMenu);
VolverAMenu.setBounds(240, 287, 192, 19);

jListTema.setBackground(new java.awt.Color(187, 83, 70));

jListTema.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.b
order.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jListTema.setCursor(new
java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
jScrollPane1.setViewportView(jListTema);

jPanel1.add(jScrollPane1);
jScrollPane1.setBounds(460, 40, 110, 131);

botonEliminarTema.setBackground(new java.awt.Color(187, 83, 70));
botonEliminarTema.setForeground(new java.awt.Color(0, 0, 0));
botonEliminarTema.setText("Eliminar Tema");

botonEliminarTema.setBorder(javax.swing.BorderFactory.createBevelBorder(javax
.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
botonEliminarTema.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        botonEliminarTemaActionPerformed(evt);
    }
});
jPanel1.add(botonEliminarTema);
botonEliminarTema.setBounds(440, 180, 150, 19);

jLabel3.setForeground(new java.awt.Color(0, 0, 0));
jLabel3.setText("Elija tema para eliminar");
jPanel1.add(jLabel3);
jLabel3.setBounds(450, 20, 150, 15);

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

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 623,
Short.MAX_VALUE)
);
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);

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

```

```

private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void jTextField2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String nombreTema = jTextField1.getText().toUpperCase();
    String descripcionTema = jTextField2.getText();
    Tema t = new Tema(nombreTema, descripcionTema);
    boolean temaExiste = existeTema(t);
    if (!temaExiste) {
        lista.add(t);
    } else {
        for (Tema tema : lista) {
            if (tema.getNombre().equals(t.getNombre())) {
                tema.setDescripcion(t.getDescripcion());
                jTextField1.setText("");
                jTextField2.setText("");
                break;
            }
        }
    }
    for (int i = 0; i < lista.size(); i++) {
        System.out.println(lista.get(i).getNombre());
    }
    setJListTema();
    jTextField1.setText("");
    jTextField2.setText("");
}

private void VolverAMenuActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose();
}

private void botonEliminarTemaActionPerformed(java.awt.event.ActionEvent
evt) {
    if (sistema.tienePreguntas((Tema) jListTema.getSelectedValue()) > 0)
    {
        JOptionPane.showMessageDialog(null, "Este tema no se puede
eliminar, tiene preguntas", "Error", JOptionPane.ERROR_MESSAGE);
    } else {
        lista.remove(jListTema.getSelectedValue());
        setJListTema();
    }
}

public boolean existeTema(Tema t) {
    boolean yaExiste = false;
    for (Tema tema : lista) {
        if (tema.getNombre().equals(t.getNombre())) {
            yaExiste = true;
            break;
        }
    }
}

```

```

        }
    }
    return yaExiste;
}

public void setJListTema() {
    jListTema.setListData(lista.toArray());
}

// Variables declaration - do not modify
private javax.swing.JButton VolverAMenu;
private javax.swing.JButton botonEliminarTema;
private javax.swing.JButton jButtonon1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JList jListTema;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
// End of variables declaration
}

```

# (JFrame) GestionTema

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Sistema;
import dominio.Tema;
import java.util.ArrayList;
import javax.swing.JOptionPane;

public class GestionTema extends javax.swing.JFrame {

    Sistema sistema;
    ArrayList<Tema> lista = new ArrayList();

    public GestionTema(ArrayList<Tema> listaTemas, Sistema sistema) {
        this.sistema = sistema;
        initComponents();
        this.setLocationRelativeTo(null);
        lista = listaTemas;
        setJListTema();
    }

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

        jPanel1 = new javax.swing.JPanel();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jButton1 = new javax.swing.JButton();
        VolverAMenu = new javax.swing.JButton();
        jScrollPane1 = new javax.swing.JScrollPane();
        jListTema = new javax.swing.JList();
        botonEliminarTema = new javax.swing.JButton();
        jLabel3 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setUndecorated(true);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));
        jPanel1.setPreferredSize(new java.awt.Dimension(600, 400));
        jPanel1.setLayout(null);

        jTextField1.setBackground(new java.awt.Color(187, 83, 70));

        jTextField1.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
        .border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
        java.awt.Color.black, java.awt.Color.black));
        jTextField1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        jTextField1ActionPerformed(evt);
    }
});
jPanel1.add(jTextField1);
jTextField1.setBounds(190, 10, 188, 19);

jTextField2.setBackground(new java.awt.Color(187, 83, 70));

jTextField2.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jTextField2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jTextField2ActionPerformed(evt);
    }
});
jPanel1.add(jTextField2);
jTextField2.setBounds(190, 90, 188, 60);

jLabel1.setBackground(new java.awt.Color(187, 83, 70));
jLabel1.setForeground(new java.awt.Color(0, 0, 0));
jLabel1.setText("Nombre del tema");
jPanel1.add(jLabel1);
jLabel1.setBounds(12, 14, 120, 15);

jLabel2.setBackground(new java.awt.Color(187, 83, 70));
jLabel2.setForeground(new java.awt.Color(0, 0, 0));
jLabel2.setText("Descripcion");
jPanel1.add(jLabel2);
jLabel2.setBounds(12, 112, 90, 15);

jButton1.setBackground(new java.awt.Color(187, 83, 70));
jButton1.setForeground(new java.awt.Color(0, 0, 0));
jButton1.setText("Guardar tema");

jButton1.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
jPanel1.add(jButton1);
jButton1.setBounds(12, 287, 130, 19);

VolverAMenu.setBackground(new java.awt.Color(187, 83, 70));
VolverAMenu.setForeground(new java.awt.Color(0, 0, 0));
VolverAMenu.setText("Volver a menu principal");

VolverAMenu.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
VolverAMenu.setCursor(new
java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
VolverAMenu.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        VolverAMenuActionPerformed(evt);
    }
});
jPanel1.add(VolverAMenu);
VolverAMenu.setBounds(240, 287, 192, 19);

jListTema.setBackground(new java.awt.Color(187, 83, 70));

jListTema.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.b
order.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
jListTema.setCursor(new
java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
jScrollPane1.setViewportView(jListTema);

jPanel1.add(jScrollPane1);
jScrollPane1.setBounds(460, 40, 110, 131);

botonEliminarTema.setBackground(new java.awt.Color(187, 83, 70));
botonEliminarTema.setForeground(new java.awt.Color(0, 0, 0));
botonEliminarTema.setText("Eliminar Tema");

botonEliminarTema.setBorder(javax.swing.BorderFactory.createBevelBorder(javax
.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
botonEliminarTema.addActionListener(new
java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        botonEliminarTemaActionPerformed(evt);
    }
});
jPanel1.add(botonEliminarTema);
botonEliminarTema.setBounds(440, 180, 150, 19);

jLabel3.setForeground(new java.awt.Color(0, 0, 0));
jLabel3.setText("Elija tema para eliminar");
jPanel1.add(jLabel3);
jLabel3.setBounds(450, 20, 150, 15);

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

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 623,
Short.MAX_VALUE)
);
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);

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

```

```

private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void jTextField2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String nombreTema = jTextField1.getText().toUpperCase();
    String descripcionTema = jTextField2.getText();
    Tema t = new Tema(nombreTema, descripcionTema);
    boolean temaExiste = existeTema(t);
    if (!temaExiste) {
        lista.add(t);
    } else {
        for (Tema tema : lista) {
            if (tema.getNombre().equals(t.getNombre())) {
                tema.setDescripcion(t.getDescripcion());
                jTextField1.setText("");
                jTextField2.setText("");
                break;
            }
        }
    }
    for (int i = 0; i < lista.size(); i++) {
        System.out.println(lista.get(i).getNombre());
    }
    setJListTema();
    jTextField1.setText("");
    jTextField2.setText("");
}

private void VolverAMenuActionPerformed(java.awt.event.ActionEvent evt) {
    this.dispose();
}

private void botonEliminarTemaActionPerformed(java.awt.event.ActionEvent
evt) {
    if (sistema.tienePreguntas((Tema) jListTema.getSelectedValue()) > 0)
    {
        JOptionPane.showMessageDialog(null, "Este tema no se puede
eliminar, tiene preguntas", "Error", JOptionPane.ERROR_MESSAGE);
    } else {
        lista.remove(jListTema.getSelectedValue());
        setJListTema();
    }
}

public boolean existeTema(Tema t) {
    boolean yaExiste = false;
    for (Tema tema : lista) {
        if (tema.getNombre().equals(t.getNombre())) {
            yaExiste = true;
            break;
        }
    }
}

```



```

        }
    }
    return yaExiste;
}

public void setJListTema() {
    jListTema.setListData(lista.toArray());
}

// Variables declaration - do not modify
private javax.swing.JButton VolverAMenu;
private javax.swing.JButton botonEliminarTema;
private javax.swing.JButton jButtonon1;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JList jListTema;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField2;
// End of variables declaration
}

```

# (JFrame) JugarMemory

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Casillero;
import dominio.Partida;
import dominio.Pregunta;
import jaco.mp3.player.MP3Player;
import java.awt.Color;
import javax.swing.*;
import java.awt.event.*;
import java.io.File;
import java.util.ArrayList;

public class JugarMemory extends javax.swing.JFrame {

    private Casillero casillero;
    private Partida partida;
    private JButton[][] botones;
    private ArrayList<Pregunta> lista;
    private Casillero c1;
    private Casillero c2;
    private int[] preguntasUsadas = {-1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
-1, -1};
    private int numeroDePregunta = 0;
    private MP3Player mp3;

    public JugarMemory(Partida partida) {
        this.partida = partida;
        initComponents();
        this.setLocationRelativeTo(null);
        int cantColumnas = 4;
        int cantFilas = 3;
        botones = new JButton[cantFilas][cantColumnas];
        String cancion = "src\\musica\\musica.mp3";
        mp3 = new MP3Player(new File(cancion));

        for (int i = 0; i < cantFilas; i++) {
            for (int j = 0; j < cantColumnas; j++) {
                JButton jButton = new JButton();
                jButton.addActionListener(new ListenerBoton(i, j));
                panelInferior.add(jButton);
                botones[i][j] = jButton;
                botones[i][j].setBackground(new Color(187, 83, 70));
                boolean agrego = false;
                while (!agrego) {
                    int numeroRandom = setNumeroRandom();
                    if (contadorPreguntas(numeroRandom) == 0) {
                        Casillero casillero = new Casillero(i, j,
partida.getListaPreguntasMezcladas().get(numeroRandom), true);
                        partida.getListaCasilleros().add(casillero);
                        agrego = true;
                        preguntasUsadas[numeroDePregunta] = numeroRandom;
                    }
                }
            }
        }
    }
}
```

```

        numeroDePregunta++;
    } else if (contadorPreguntas(numeroRandom) == 1) {
        Casillero casillero = new Casillero(i, j,
partida.getListaPreguntasMezcladas().get(numeroRandom), false);
        partida.getListaCasilleros().add(casillero);
        agrego = true;
        preguntasUsadas[numeroDePregunta] = numeroRandom;
        numeroDePregunta++;
    } else if (contadorPreguntas(numeroRandom) == 2) {
        agrego = false;
    }
}
}
}
}

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

    panelSuperior = new javax.swing.JPanel();
    btnAyudaParcial = new javax.swing.JButton();
    btnAyudaTotal = new javax.swing.JButton();
    btnReset = new javax.swing.JButton();
    btnSonido = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();
    nuestroPuntaje = new javax.swing.JLabel();
    btnSalir = new javax.swing.JButton();
    panelInferior = new javax.swing.JPanel();

    setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
    setUndecorated(true);
    setPreferredSize(new java.awt.Dimension(1600, 1400));
    getContentPane().setLayout(new
javax.swing.BoxLayout(getContentPane(), javax.swing.BoxLayout.Y_AXIS));

    panelSuperior.setBackground(new java.awt.Color(255, 255, 255));
    panelSuperior.setPreferredSize(new java.awt.Dimension(600, 400));
    panelSuperior.setLayout(null);

    btnAyudaParcial.setBackground(new java.awt.Color(187, 83, 70));
    btnAyudaParcial.setForeground(new java.awt.Color(0, 0, 0));
    btnAyudaParcial.setText("Ayuda Parcial");

    btnAyudaParcial.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.s
wing.border.BevelBorder.RAISED, new java.awt.Color(0, 0, 0), new
java.awt.Color(0, 0, 0), new java.awt.Color(0, 0, 0), new java.awt.Color(0,
0, 0)));
    btnAyudaParcial.addActionListener(new java.awt.event.ActionListener()
{
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnAyudaParcialActionPerformed(evt);
    }
});
    panelSuperior.add(btnAyudaParcial);
    btnAyudaParcial.setBounds(280, 0, 80, 30);

```

```

        btnAyudaTotal.setBackground(new java.awt.Color(187, 83, 70));
        btnAyudaTotal.setForeground(new java.awt.Color(0, 0, 0));
        btnAyudaTotal.setText("Ayuda Total");

        btnAyudaTotal.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
        btnAyudaTotal.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnAyudaTotalActionPerformed(evt);
            }
        });
        panelSuperior.add(btnAyudaTotal);
        btnAyudaTotal.setBounds(360, 0, 80, 30);

        btnReset.setBackground(new java.awt.Color(187, 83, 70));
        btnReset.setForeground(new java.awt.Color(0, 0, 0));
        btnReset.setText("Reset");

        btnReset.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
        btnReset.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnResetActionPerformed(evt);
            }
        });
        panelSuperior.add(btnReset);
        btnReset.setBounds(440, 0, 80, 30);

        btnSonido.setBackground(new java.awt.Color(187, 83, 70));
        btnSonido.setForeground(new java.awt.Color(0, 0, 0));
        btnSonido.setText("Sonido");

        btnSonido.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
        btnSonido.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnSonidoActionPerformed(evt);
            }
        });
        panelSuperior.add(btnSonido);
        btnSonido.setBounds(520, 0, 80, 30);

        jLabel1.setForeground(new java.awt.Color(0, 0, 0));
        jLabel1.setText("Puntos:");
        panelSuperior.add(jLabel1);
        jLabel1.setBounds(620, 0, 50, 30);

        nuestroPuntaje.setForeground(new java.awt.Color(0, 0, 0));
        nuestroPuntaje.setText("-----");
        nuestroPuntaje.setVerticalAlignment(javax.swing.SwingConstants.TOP);
        panelSuperior.add(nuestroPuntaje);
        nuestroPuntaje.setBounds(670, 10, 50, 30);

```

```

        btnSalir.setBackground(new java.awt.Color(187, 83, 70));
        btnSalir.setForeground(new java.awt.Color(0, 0, 0));
        btnSalir.setText("Salir");

        btnSalir.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
        btnSalir.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnSalirActionPerformed(evt);
            }
        });
        panelSuperior.add(btnSalir);
        btnSalir.setBounds(200, 0, 80, 30);

        getContentPane().add(panelSuperior);

        panelInferior.setBackground(new java.awt.Color(255, 255, 255));
        panelInferior.setForeground(new java.awt.Color(187, 83, 70));
        panelInferior.setPreferredSize(new java.awt.Dimension(1400, 1000));
        panelInferior.setLayout(new java.awt.GridLayout(3, 4));
        getContentPane().add(panelInferior);

        setBounds(0, 0, 899, 463);
    } // </editor-fold>

    private void btnSonidoActionPerformed(java.awt.event.ActionEvent evt) {
        if (mp3.isStopped()) {
            mp3.play();
        } else {
            mp3.stop();
        }
    }

    private void btnAyudaParcialActionPerformed(java.awt.event.ActionEvent
evt) {
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 4; j++) {
                for (int k = 0; k < partida.getListCasilleros().size(); k++)
                {
                    Casillero c = partida.getListCasilleros().get(k);
                    if (c.getPosicionX() == i && c.getPosicionY() == j &&
c.getEsPregunta() && !c.isUsado()) {
                        botones[i][j].setText("Pregunta");
                    } else if (c.getPosicionX() == i && c.getPosicionY() == j
&& !c.getEsPregunta() && !c.isUsado()) {
                        botones[i][j].setText("Respuesta");
                    }
                }
            }
        }
    }

    private void btnResetActionPerformed(java.awt.event.ActionEvent evt) {
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 4; j++) {

```

```

        for (int k = 0; k < partida.getListCasilleros().size(); k++)
    {
        Casillero c = partida.getListCasilleros().get(k);
        if (c.getPosicionX() == i && c.getPosicionY() == j &&
!c.isUsado()) {
            botones[i][j].setText("");
        }
    }
}

private void btnAyudaTotalActionPerformed(java.awt.event.ActionEvent evt)
{
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 4; j++) {
            for (int k = 0; k < partida.getListCasilleros().size(); k++)
            {
                Casillero c = partida.getListCasilleros().get(k);
                if (c.getPosicionX() == i && c.getPosicionY() == j &&
c.getEsPregunta() && !c.isUsado()) {
                    botones[i][j].setText(c.getPregunta().getPregunta());
                } else if (c.getPosicionX() == i && c.getPosicionY() == j
&& !c.getEsPregunta() && !c.isUsado()) {
                    botones[i][j].setText(c.getPregunta().getRespuesta());
                }
            }
        }
    }

    private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
        this.dispose();
    }

    private class ListenerBoton implements ActionListener {

        private int x;
        private int y;

        public ListenerBoton(int i, int j) {
            x = i;
            y = j;
        }

        public void actionPerformed(ActionEvent e) {
            clickBoton(x, y);
        }
    }

    private void clickBoton(int fila, int columna) {
        if (!preguntaYaUsada(fila, columna)) {
            if (partida.getTurno() % 2 == 0) {
                partida.setTurno(partida.getTurno() + 1);
            }
        }
    }
}

```

```

        for (int i = 0; i < partida.getListCasilleros().size(); i++)
    {
        if (fila ==
partida.getListCasilleros().get(i).getPosicionX() && columna ==
partida.getListCasilleros().get(i).getPosicionY()) {
            c1 = partida.getListCasilleros().get(i);
            break;
        }
    }
    if (c1.getEsPregunta()) {

botones[fila][columna].setText(c1.getPregunta().getPregunta());
        } else if (!c1.getEsPregunta()) {

botones[fila][columna].setText(c1.getPregunta().getRespuesta());
        }
    } else if (partida.getTurno() % 2 == 1) {
        partida.setTurno(partida.getTurno() + 1);

        if (partida.getTurno() % 2 == 0) {
            for (int i = 0; i < partida.getListCasilleros().size();
i++) {
                if (fila ==
partida.getListCasilleros().get(i).getPosicionX() && columna ==
partida.getListCasilleros().get(i).getPosicionY()) {
                    c2 = partida.getListCasilleros().get(i);
                    break;
                }
            }
            if (c2.getEsPregunta()) {

botones[fila][columna].setText(c2.getPregunta().getPregunta());
                } else if (!c2.getEsPregunta()) {

botones[fila][columna].setText(c2.getPregunta().getRespuesta());
                }
            if (c1.getPregunta().getPregunta() ==
c2.getPregunta().getPregunta() && c1 != c2) {
                JOptionPane.showMessageDialog(null, "le embocaste",
"Error", JOptionPane.ERROR_MESSAGE);
                partida.setPuntajeJugador(partida.getPuntajeJugador()
+ 50);

muestroPuntaje.setText(String.valueOf(partida.getPuntajeJugador()));
                c1.setUsado(true);
                c2.setUsado(true);
            } else {
                partida.setPuntajeJugador(partida.getPuntajeJugador()
- 10);

muestroPuntaje.setText(String.valueOf(partida.getPuntajeJugador()));
                JOptionPane.showMessageDialog(null, "le erraste",
"Error", JOptionPane.ERROR_MESSAGE);

botones[c1.getPosicionX()][c1.getPosicionY()].setText("");

botones[c2.getPosicionX()][c2.getPosicionY()].setText("");

```

```

        }
    }
}

} else {
    JOptionPane.showMessageDialog(null, "Ya usaste este boton",
    "Error", JOptionPane.ERROR_MESSAGE);
}
}

public boolean preguntaYaUsada(int fila, int columna) {
    boolean usado = false;
    for (int i = 0; i < partida.getListCasilleros().size(); i++) {
        if (fila == partida.getListCasilleros().get(i).getPosicionX() &&
columna == partida.getListCasilleros().get(i).getPosicionY()) {
            usado = partida.getListCasilleros().get(i).isUsado();
        }
    }
    return usado;
}

public int setNumeroRandom() {
    int num = (int) (Math.floor(Math.random() * 6));
    return num;
}

public int contadorPreguntas(int numero) {
    int contador = 0;
    for (int i = 0; i < preguntasUsadas.length; i++) {
        if (numero == preguntasUsadas[i]) {
            contador++;
        }
    }
    return contador;
}

// Variables declaration - do not modify
private javax.swing.JButton btnAyudaParcial;
private javax.swing.JButton btnAyudaTotal;
private javax.swing.JButton btnReset;
private javax.swing.JButton btnSalir;
private javax.swing.JButton btnSonido;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel nuestroPuntaje;
private javax.swing.JPanel panelInferior;
private javax.swing.JPanel panelSuperior;
// End of variables declaration
}

```



# (JFrame) MenuPrincipal

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Sistema;
import obligatorio2p2.Obligatorio2P2;

public class MenuPrincipal extends javax.swing.JFrame {

    private Sistema sistema;

    public MenuPrincipal(Sistema unSistema) {
        initComponents();
        this.sistema = unSistema;
        setTitle("Bienvenido");
    }

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

        jPanel1 = new javax.swing.JPanel();
        GestionarTemas = new javax.swing.JButton();
        GestionarPreguntasManualmente = new javax.swing.JButton();
        botonJugar = new javax.swing.JButton();
        btnSalir = new javax.swing.JButton();
        guardarDatos = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.DO_NOTHING_ON_CLOSE);
        setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
        setUndecorated(true);
        setResizable(false);
        setSize(new java.awt.Dimension(600, 400));

        jPanel1.setCursor(new
        java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
        jPanel1.setPreferredSize(new java.awt.Dimension(700, 600));
        jPanel1.setLayout(null);

        GestionarTemas.setBackground(new java.awt.Color(187, 83, 70));
        GestionarTemas.setText("Gestionar Temas");

        GestionarTemas.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.s
        ing.border.BevelBorder.LOWERED, java.awt.Color.black, java.awt.Color.black,
        java.awt.Color.black, java.awt.Color.black));
        GestionarTemas.setCursor(new
        java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
        GestionarTemas.addActionListener(new java.awt.event.ActionListener()
        {

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

```

    }
    });
    jPanell1.add(GestionarTemas);
    GestionarTemas.setBounds(10, 10, 210, 60);

    GestionarPreguntasManualmente.setBackground(new java.awt.Color(187,
83, 70));
    GestionarPreguntasManualmente.setText("Gestionar Preguntas
Manualmente");

    GestionarPreguntasManualmente.setBorder(javax.swing.BorderFactory.createBevel
Border(javax.swing.border.BevelBorder.LOWERED, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
    GestionarPreguntasManualmente.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            GestionarPreguntasManualmenteActionPerformed(evt);
        }
    });
    jPanell1.add(GestionarPreguntasManualmente);
    GestionarPreguntasManualmente.setBounds(10, 80, 210, 60);

    botonJugar.setBackground(new java.awt.Color(187, 83, 70));
    botonJugar.setText("Jugar");

    botonJugar.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.
border.BevelBorder.LOWERED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
    botonJugar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            botonJugarActionPerformed(evt);
        }
    });
    jPanell1.add(botonJugar);
    botonJugar.setBounds(540, 350, 90, 19);

    btnSalir.setBackground(new java.awt.Color(187, 83, 70));
    btnSalir.setText("salir");

    btnSalir.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.LOWERED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
    btnSalir.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnSalirActionPerformed(evt);
        }
    });
    jPanell1.add(btnSalir);
    btnSalir.setBounds(230, 350, 90, 19);

    guardarDatos.setBackground(new java.awt.Color(187, 83, 70));
    guardarDatos.setText("Opciones Generacion Datos");

    guardarDatos.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swin
g.border.BevelBorder.RAISED, new java.awt.Color(0, 0, 0), new
java.awt.Color(0, 0, 0), new java.awt.Color(0, 0, 0), new java.awt.Color(0,
0, 0)));

```

```

        guardarDatos.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                guardarDatosActionPerformed(evt);
            }
        });
        jPanel1.add(guardarDatos);
        guardarDatos.setBounds(10, 150, 210, 60);

        jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Imágenes/cerebro.png"))); //
NOI18N
        jLabel1.setText("jLabel1");
        jPanel1.add(jLabel1);
        jLabel1.setBounds(0, -20, 760, 460);

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

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 0, Short.MAX_VALUE))
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE, 417,
Short.MAX_VALUE))
        );

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

    private void GestionarTemasActionPerformed(java.awt.event.ActionEvent
evt) {
        GestionTema gestionT = new GestionTema(sistema.getListasTemas(),
sistema);
        gestionT.setVisible(true);
    }

    private void
GestionarPreguntasManualmenteActionPerformed(java.awt.event.ActionEvent evt)
{
        GestionPregunta gestionP = new GestionPregunta(sistema);
        gestionP.setVisible(true);
    }

    private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
        Obligatorio2P2.guardarSistema(this.sistema);
        this.dispose();
    }

```

```

private void botonJugarActionPerformed(java.awt.event.ActionEvent evt) {
    SeleccionarTemas select = new SeleccionarTemas(sistema);
    select.setVisible(true);
}

private void guardarDatosActionPerformed(java.awt.event.ActionEvent evt)
{
    CargaDatosVentanaAuxiliar c = new CargaDatosVentanaAuxiliar(sistema);
    c.setVisible(true);
}

// Variables declaration - do not modify
private javax.swing.JButton GestionarPreguntasManualmente;
private javax.swing.JButton GestionarTemas;
private javax.swing.JButton botonJugar;
private javax.swing.JButton btnSalir;
private javax.swing.JButton guardarDatos;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
// End of variables declaration
}

```

# (JFrame) ModoFlashCard

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Partida;
import java.util.Collection;
import javax.swing.JOptionPane;

public class ModoFlashCard extends javax.swing.JFrame {

    private Partida partida;
    int numero = 0;

    public ModoFlashCard(Partida partida) {
        this.partida = partida;
        initComponents();
        if (partida.getListaPreguntasElegidas().size() > 0) {

            btnTexto.setText(partida.getListaPreguntasElegidas().get(numero).getPregunta(
            ));
        } else {
            JOptionPane.showMessageDialog(null, "No hay preguntas cargadas",
            "Error", JOptionPane.ERROR_MESSAGE);
        }
    }

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

        jPanel1 = new javax.swing.JPanel();
        btnPreguntaAnterior = new javax.swing.JButton();
        btnSiguientePregunta = new javax.swing.JButton();
        btnTexto = new javax.swing.JButton();
        btnSalir = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.DO_NOTHING_ON_CLOSE);
        setUndecorated(true);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));
        jPanel1.setPreferredSize(new java.awt.Dimension(415, 200));
        jPanel1.setLayout(null);

        btnPreguntaAnterior.setBackground(new java.awt.Color(187, 83, 70));
        btnPreguntaAnterior.setForeground(new java.awt.Color(0, 0, 0));
        btnPreguntaAnterior.setText("Anterior");

        btnPreguntaAnterior.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black,
        java.awt.Color.black, java.awt.Color.black, new java.awt.Color(0, 0, 0)));
        btnPreguntaAnterior.addActionListener(new
        java.awt.event.ActionListener() {
```

```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnPreguntaAnteriorActionPerformed(evt);
        }
    });
    jPanel1.add(btnPreguntaAnterior);
    btnPreguntaAnterior.setBounds(12, 55, 60, 30);

    btnSiguientePregunta.setBackground(new java.awt.Color(187, 83, 70));
    btnSiguientePregunta.setForeground(new java.awt.Color(0, 0, 0));
    btnSiguientePregunta.setText("Siguiente");

    btnSiguientePregunta.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
    btnSiguientePregunta.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnSiguientePreguntaActionPerformed(evt);
        }
    });
    jPanel1.add(btnSiguientePregunta);
    btnSiguientePregunta.setBounds(323, 55, 60, 30);

    btnTexto.setBackground(new java.awt.Color(187, 83, 70));

    btnTexto.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
    btnTexto.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnTextoActionPerformed(evt);
        }
    });
    jPanel1.add(btnTexto);
    btnTexto.setBounds(99, 27, 213, 128);

    btnSalir.setBackground(new java.awt.Color(187, 83, 70));
    btnSalir.setForeground(new java.awt.Color(0, 0, 0));
    btnSalir.setText("Salir");

    btnSalir.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, new java.awt.Color(0, 0, 0), java.awt.Color.black));
    btnSalir.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnSalirActionPerformed(evt);
        }
    });
    jPanel1.add(btnSalir);
    btnSalir.setBounds(0, 159, 90, 30);

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

        .addComponent(jPanell1,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanell1,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    );

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

private void btnTextoActionPerformed(java.awt.event.ActionEvent evt) {
    if (partida.getListaPreguntasElegidas().size() > 0) {
        if (btnTexto.getText() ==
partida.getListaPreguntasElegidas().get(numero).getRespuesta()) {

btnTexto.setText(partida.getListaPreguntasElegidas().get(numero).getPregunta(
));
        } else {

btnTexto.setText(partida.getListaPreguntasElegidas().get(numero).getRespuesta(
));
        }
    }
}

private void
btnSiguientePreguntaActionPerformed(java.awt.event.ActionEvent evt) {
    if (partida.getListaPreguntasElegidas().size() > 1) {
        if (numero == partida.getListaPreguntasElegidas().size() - 1) {
            numero = 0;
        } else {
            numero++;
        }

btnTexto.setText(partida.getListaPreguntasElegidas().get(numero).getPregunta(
));
    }
}

private void
btnPreguntaAnteriorActionPerformed(java.awt.event.ActionEvent evt) {
    if (partida.getListaPreguntasElegidas().size() > 1) {
        if (numero == 0) {
            numero = partida.getListaPreguntasElegidas().size() - 1;
        } else {
            numero--;
        }

btnTexto.setText(partida.getListaPreguntasElegidas().get(numero).getPregunta(
));
    }
}

```

```
}  
private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {  
    partida.getListaPreguntasElegidas().clear();  
    partida.getListaTemasElegidos().clear();  
    this.dispose();  
}  
  
// Variables declaration - do not modify  
private javax.swing.JButton btnPreguntaAnterior;  
private javax.swing.JButton btnSalir;  
private javax.swing.JButton btnSiguientePregunta;  
private javax.swing.JButton btnTexto;  
private javax.swing.JPanel jPanel1;  
// End of variables declaration  
}
```



# (JFrame) SeleccionarTemas

```
//Nombres y numeros de estudiantes
//Ivan kaitazoff: 233940
//Sebastian Romelli: 222405
package interfaz;

import dominio.Partida;
import dominio.Pregunta;
import dominio.Sistema;
import dominio.Tema;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
import javax.swing.JOptionPane;

public class SeleccionarTemas extends javax.swing.JFrame {

    private Sistema sist;
    private Partida partida;

    public SeleccionarTemas(Sistema sistema) {

        initComponents();
        this.sist = sistema;
        ArrayList<Tema> temas = sist.getListasTemas();
        jListaEleccionTemas.setListData(temas.toArray());
        ArrayList<Tema> temasElegidosMenu = new ArrayList<>();
        partida = new Partida(temasElegidosMenu, 0, 0);

    }

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

        jPanel1 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jScrollPane1 = new javax.swing.JScrollPane();
        jListaTemasElegidos = new javax.swing.JList();
        jLabel2 = new javax.swing.JLabel();
        botonSeleccionarTema = new javax.swing.JToggleButton();
        comenzarPartida = new javax.swing.JToggleButton();
        modoFlashCard = new javax.swing.JToggleButton();
        jScrollPane2 = new javax.swing.JScrollPane();
        jListaEleccionTemas = new javax.swing.JList();
        btnSalir = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
        setUndecorated(true);

        jPanel1.setBackground(new java.awt.Color(255, 255, 255));
        jPanel1.setPreferredSize(new java.awt.Dimension(600, 400));
        jPanel1.setLayout(null);
```

```

jLabel1.setForeground(new java.awt.Color(0, 0, 0));
jLabel1.setText("Selecciones el o los temas a utilizar ");
jPanel1.add(jLabel1);
jLabel1.setBounds(37, 12, 174, 15);

jListaTemasElegidos.setBackground(new java.awt.Color(187, 83, 70));

jListaTemasElegidos.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
jScrollPane1.setViewportView(jListaTemasElegidos);

jPanel1.add(jScrollPane1);
jScrollPane1.setBounds(378, 33, 200, 230);

jLabel2.setForeground(new java.awt.Color(0, 0, 0));
jLabel2.setText("Temas Seleccionados");
jPanel1.add(jLabel2);
jLabel2.setBounds(428, 12, 106, 15);

botonSeleccionarTema.setBackground(new java.awt.Color(187, 83, 70));
botonSeleccionarTema.setForeground(new java.awt.Color(0, 0, 0));
botonSeleccionarTema.setText("Seleccionar tema");

botonSeleccionarTema.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
botonSeleccionarTema.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        botonSeleccionarTemaActionPerformed(evt);
    }
});
jPanel1.add(botonSeleccionarTema);
botonSeleccionarTema.setBounds(30, 280, 170, 19);

comenzarPartida.setBackground(new java.awt.Color(187, 83, 70));
comenzarPartida.setForeground(new java.awt.Color(0, 0, 0));
comenzarPartida.setText("Jugar memory");

comenzarPartida.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
comenzarPartida.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        comenzarPartidaActionPerformed(evt);
    }
});
jPanel1.add(comenzarPartida);
comenzarPartida.setBounds(480, 349, 100, 30);

modoFlashCard.setBackground(new java.awt.Color(187, 83, 70));
modoFlashCard.setForeground(new java.awt.Color(0, 0, 0));
modoFlashCard.setText("Jugar modo flash cards");

modoFlashCard.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swi

```

```

ng.border.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black, new
java.awt.Color(0, 0, 0), java.awt.Color.black));
    modoFlashCard.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            modoFlashCardActionPerformed(evt);
        }
    });
    jPanel1.add(modoFlashCard);
    modoFlashCard.setBounds(270, 349, 190, 30);

    jListaEleccionTemas.setBackground(new java.awt.Color(187, 83, 70));

    jListaEleccionTemas.setBorder(javax.swing.BorderFactory.createBevelBorder(javax
.swing.border.BevelBorder.RAISED, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black, java.awt.Color.black));
    jScrollPane2.setViewportView(jListaEleccionTemas);

    jPanel1.add(jScrollPane2);
    jScrollPane2.setBounds(20, 30, 200, 230);

    btnSalir.setBackground(new java.awt.Color(187, 83, 70));
    btnSalir.setForeground(new java.awt.Color(0, 0, 0));
    btnSalir.setText("Salir");

    btnSalir.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED, java.awt.Color.black, java.awt.Color.black,
java.awt.Color.black, java.awt.Color.black));
    btnSalir.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnSalirActionPerformed(evt);
        }
    });
    jPanel1.add(btnSalir);
    btnSalir.setBounds(10, 349, 90, 30);

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

        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );
    layout.setVerticalGroup(

        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );

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

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

```

```

        Tema temaS = (Tema) jListaEleccionTemas.getSelectedValue();
        if (!partida.getListaTemasElegidos().contains(temaS)) {
            agregarTema(sist, temaS);
        } else {
            JOptionPane.showMessageDialog(null, "tema ya seleccionado",
"Error", JOptionPane.ERROR_MESSAGE);
        }

    }

    private void comenzarPartidaActionPerformed(java.awt.event.ActionEvent
evt) {
        partida.cargarPreguntasPartida(sist.getListaPreguntas());
        List<Pregunta> preguntasMezcladas =
partida.getListaPreguntasElegidas();
        if (partida.getListaPreguntasElegidas().size() >= 6) {
            Collections.shuffle(preguntasMezcladas);
            partida.setListaPreguntasMezcladas((ArrayList<Pregunta>)
preguntasMezcladas);
            JugarMemory memory = new JugarMemory(partida);
            memory.setVisible(true);
            this.dispose();
        } else {
            JOptionPane.showMessageDialog(null, "Debe haber al menos 6
preguntas cargadas para que comience la partida", "Error",
JOptionPane.ERROR_MESSAGE);
        }

    }

    private void modoFlashCardActionPerformed(java.awt.event.ActionEvent evt)
{
        partida.cargarPreguntasPartida(sist.getListaPreguntas());
        this.dispose();
        ModoFlashCard flashCards = new ModoFlashCard(partida);
        flashCards.setVisible(true);

    }

    private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
        this.dispose();
    }

    public Partida crearPartida(Sistema sist, ArrayList<Tema>
listaTemasElegidos) {

        Partida partida = new Partida(listaTemasElegidos, 0, 0);
        partida.cargarPreguntasPartida(sist.getListaPreguntas());

        return partida;
    }

    public void agregarTema(Sistema sist, Tema tema) {
        partida.getListaTemasElegidos().add(tema);

jListaTemasElegidos.setListData(partida.getListaTemasElegidos().toArray());

```

```
}

// Variables declaration - do not modify
private javax.swing.JToggleButton botonSeleccionarTema;
private javax.swing.JButton btnSalir;
private javax.swing.JToggleButton comenzarPartida;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JList jListaEleccionTemas;
private javax.swing.JList jListaTemasElegidos;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JToggleButton modoFlashCard;
// End of variables declaration
}
```