

UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA

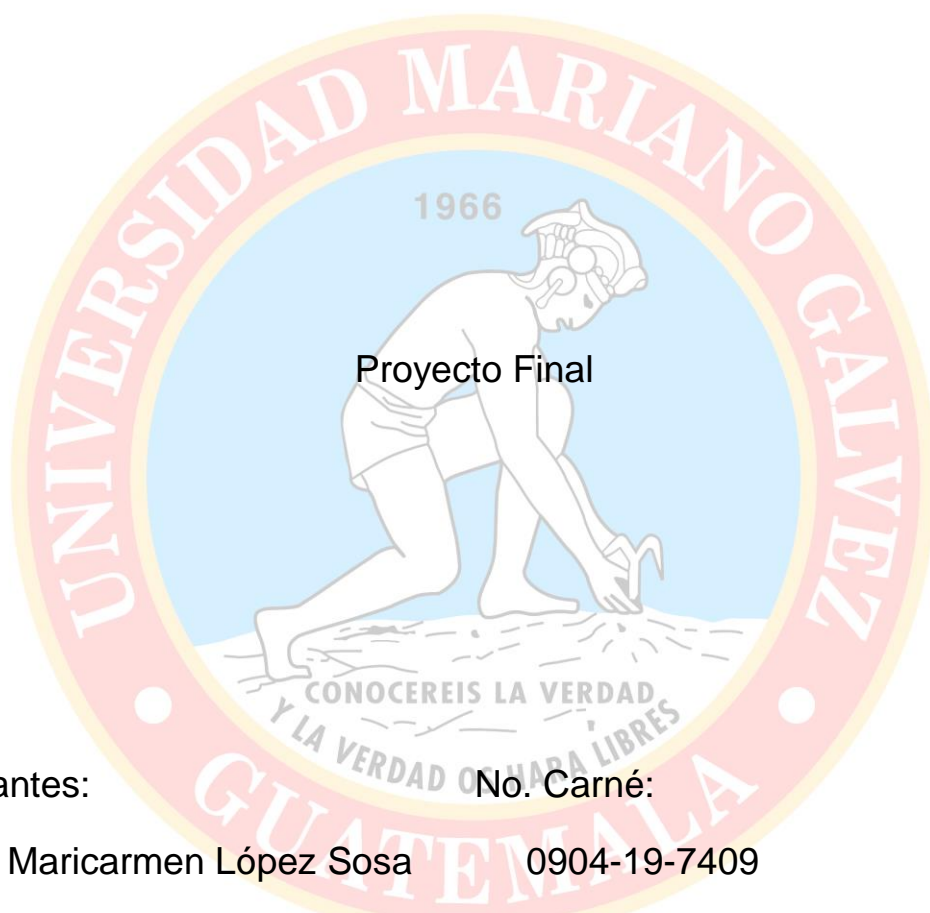
CAMPUS DE HUEHUETENANGO

Curso: Programación II

Catedrático: Ing. Johnny Morales

Plan: sábado

Sección: "A"



Estudiantes:

No. Carné:

Evelin Maricarmen López Sosa

0904-19-7409

Esvin Geovanny Figueroa Figueró

0904-19-3400

Octubre de 2020

INDICE

INTRODUCCIÓN.....	3
PROCESO DE ELABORACIÓN	4
TECNOLOGIAS UTILIZADAS.....	10
DIAGRAMA DE CLASES.....	11
ER FÍSICO.....	13
MANUAL DE USUARIO	14
CONCLUSIONES.....	16

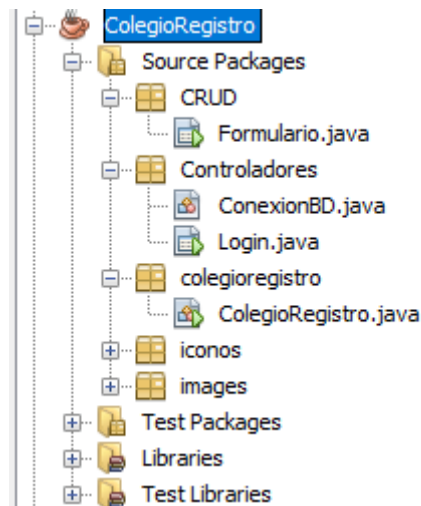
INTRODUCCIÓN

Java es un lenguaje de programación y una plataforma informática que fue comercializada por primera vez en 1995 por Sun Microsystems. Java es rápido, seguro y fiable. Por otro lado, la programación Orientada a Objetos es un paradigma que viene a innovar la forma de obtener resultados. Los objetos se utilizan como metáfora para emular las entidades reales del negocio a modelar.

Nuestro proyecto final fue realizado con programación Orientada a Objetos, ya que consistía en un Control de alumnos, fue programado en NetBeans que es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java, y persistimos los datos utilizando una base de datos la cual es MySQL el cual es un sistema de gestión de bases de datos relacional y está considerada como la base de datos de código abierto más popular del mundo.

PROCESO DE ELABORACIÓN

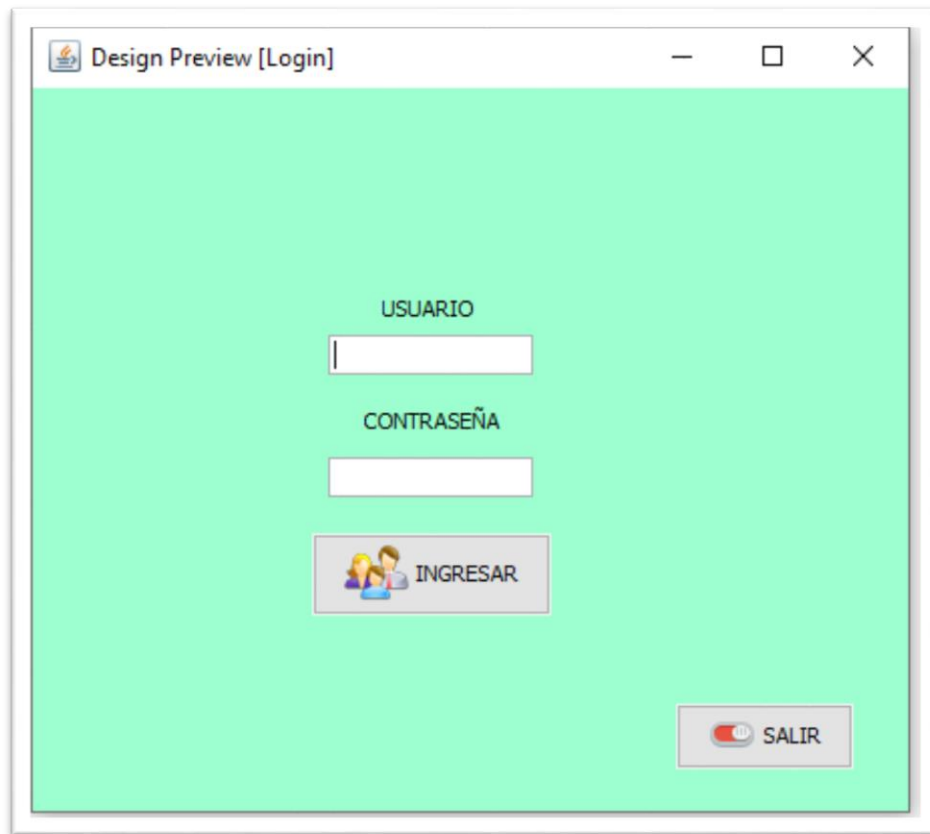
ESTRUCTURA DE PROYECTO



Clase ColegioRegistro (MAIN)

```
art Page x | Formulario.java x | ConexionBD.java x | Login.java x | ColegioRegistro.java x
source History | [Icons] | [Tools] | [Run] | [Debug] | [Test] | [Build] | [View] | [Window] | [Help]
1
2 package colegioregistro;
3
4
5 import Controladores.ConexionBD;
6 import Controladores.Login;
7 import java.sql.SQLException;
8
9
10
11 public class ColegioRegistro {
12
13     public static void main(String[] args) throws SQLException {
14
15         ConexionBD co = new ConexionBD();
16         co.getConexion();
17
18         if(co.cadena!=null){
19             System.out.println("EXITO");
20             System.out.println(co.cadena);
21
22             Login login = new Login();
23             login.setVisible(true);
24         } else {
25             System.out.println("No hay conexion");
26         }
27     }
28 }
```

LOGIN



Código Clase Login

The screenshot shows an IDE with several open files: Formulario.java, ConexionBD.java, Login.java, and ColegioRegistro.java. The 'Login.java' file is active, displaying the following Java code:

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

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

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    dispose();
}

public void validar(){

    String usuario = txtUsuario.getText().trim();
    String contra = txtPassword.getText().trim();

    if (usuario.equals("Esvin") || usuario.equals("Evelin") && contra.equals("12345"))
    {
        Formulario registro = new Formulario();
        registro.setVisible(true);
    }
    else
    {
        JOptionPane.showMessageDialog(this, "Credenciales Incorrectas", "Error de Bloqueo", JOptionPane.WARNING_MESSAGE);
    }
}
```

Clase Conexión a Base de Datos

```
2 package Controladores;
3
4 import java.sql.Connection;
5 import java.sql.DriverManager;
6
7 public class ConexionBD {
8
9     private final String DRIVER = "com.mysql.jdbc.Driver";
10    private final String URL = "jdbc:mysql://localhost:3306/colegio";
11    private final String USER = "root";
12    private final String PASSWORD = "12345";
13
14    public Connection cadena;
15    public Connection getConexion;
16
17    public ConexionBD(){
18        this.cadena = null;
19    }
20
21    public Connection getConexion(){
22        try{
23            Class.forName(DRIVER);
24            this.cadena = DriverManager.getConnection(URL, USER, PASSWORD);
25        } catch (Exception e) {
26            System.out.println(e.getMessage());
27        }
28        return this.cadena;
29    }
30
31    public void desconectar(){
32        try{
33            this.cadena.close();
34        } catch (Exception e) {
35            System.out.println(e.getMessage());
36        }
37    }
38 }
```

Formulario Registro

Design Preview [Formulario]

REGISTRO




Datos

ID: PRIMER APELLIDO

NOMBRE SEGUNDO APELLIDO

SEGUNDO NOMBRE EDAD

Operaciones

 Agregar  Modificar  Eliminar  Nuevo  REGRESAR

Lista

ID	PRIMER_NOMBRE	SEGUNDO_NOMBRE	PRIMER_APELLIDO	SEGUNDO_APELLIDO	EDAD
----	---------------	----------------	-----------------	------------------	------

Código tabla

```
private void TablaDatosMouseClicked(java.awt.event.MouseEvent evt) {  
    int row = TablaDatos.getSelectedRow();  
    if (row == -1) {  
        JOptionPane.showMessageDialog(null, "No se Selecciono");  
    } else {  
        id = Integer.parseInt((String) TablaDatos.getValueAt(row, 0).toString());  
        String nombre = (String) TablaDatos.getValueAt(row, 1);  
        String snombre = (String) TablaDatos.getValueAt(row, 2);  
        String apelli = (String) TablaDatos.getValueAt(row, 3);  
        String sapell = (String) TablaDatos.getValueAt(row, 4);  
        String edad = (String) TablaDatos.getValueAt(row, 5);  
  
        txtId.setText("" + id);  
        txtNom.setText(nombre);  
        txtSnombre.setText(snombre);  
        txtApellido.setText(apelli);  
        txtSapellido.setText(sapell);  
        txtEdad.setText(edad);  
    }  
}
```

Código Listar

```
void listar() {
    String query = "select * from alumno";
    try {
        con = cn.getConnection();
        st = con.createStatement();
        rs = st.executeQuery(query);
        Object[] alumno = new Object[6];
        String[] Titulos = {"ID", "PRIMER NOMBRE", "SEGUNDO NOMBRE", "PRIMER APELLIDO", "SEGUNDO APELLIDO", "EDAD"};
        model = new DefaultTableModel(null, Titulos);
        model = (DefaultTableModel) TablaDatos.getModel();
        while (rs.next()) {
            alumno[0] = rs.getInt("Id");
            alumno[1] = rs.getString("PRIMER NOMBRE");
            alumno[2] = rs.getString("SEGUNDO NOMBRE");
            alumno[3] = rs.getString("PRIMER APELLIDO");
            alumno[4] = rs.getString("SEGUNDO APELLIDO");
            alumno[5] = rs.getString("EDAD");
            model.addRow(alumno);
        }
        TablaDatos.setModel(model);
    } catch (Exception e) {
    }
}
```

Código Agregar

```
] void Agregar() {
    //Instancia de la clase conexión//
    ConexionBD conexion = new ConexionBD();
    //Se crea una nueva variable, invocando la conexión//
    Connection cone = conexion.getConexion();
    //Se crean las variables para el query y el resultset
    PreparedStatement ps = null;
    //String dni = txtNombre.getText();
    String nombre = txtNom.getText();
    String snombre = txtSnombre.getText();
    String apelli = txtApellido.getText();
    String sapelli = txtSapellido.getText();
    String edad = txtEdad.getText();
    // Declaración del query

    //Fin declaración query
    if (nombre.equals("") || apelli.equals("") || edad.equals("")) {
        JOptionPane.showMessageDialog(null, "Debe Ingresar Datos");
        limpiarTabla(model);
    } else {
```



```

    } else {

        try {
            //Se genera el query para la inserción de datos
            ps = cone.prepareStatement("INSERT INTO `colegio`.`alumno` (`nombre_alumno`, `segundo_nombre`, "
                                     + "`primer_apellido`, `segundo_apellido`, `edad`) VALUES (?, ?, ?, ?, ?)");
            ps.setString(1, nombre);
            ps.setString(2, snombre);
            ps.setString(3, apelli);
            ps.setString(4, sapelli);
            ps.setInt(5, Integer.parseInt(edad));
            //Se ejecuta la consulta
            ps.execute();

        } catch (SQLException e) {
            System.out.println("error " + e);
        }
    }
}

```

Código Modificar

```

void Modificar() {
    //String dni = txtCARNE.getText();
    String snombre = txtSnombre.getText();
    String nombre = txtNom.getText();
    String apelli = txtApellido.getText();
    String sapelli = txtSapellido.getText();
    String edad = txtEdad.getText();

    String query = "update alumno set NOMBRE='" + nombre + "', Segundo Nombres='" + snombre + "', ,where Id=" + id;
    try {
        if (nombre != null || snombre != null) {
            con = cn.getConnection();
            st = con.createStatement();
            st.executeUpdate(query);
            JOptionPane.showMessageDialog(null, "Usuario Modificado");
            limpiarTabla(model);
        } else {
            JOptionPane.showMessageDialog(null, "Error...!!!");
        }
    } catch (Exception e) {
    }
}

```

Código Eliminar

```
void Eliminar() {
    String query = "delete from alumno where Id=" + id;
    int fila = TablaDatos.getSelectedRow();
    if (fila < 0) {
        JOptionPane.showMessageDialog(null, "Usuario no Seleccionado");
    } else {
        try {
            con = cn.getConnection();
            st = con.createStatement();
            st.executeUpdate(query);
            JOptionPane.showMessageDialog(null, "Alumno Eliminado");
            limpiarTabla(model);
        } catch (Exception e) {
        }
    }
}
```

Código Nuevo, LimpiarTabla, Actualizar

```
void nuevo() {
    txtId.setText("");
    // txtCARNE.setText("");
    txtNom.setText("");
    //txtCARNE.requestFocus();
    txtSnombre.setText("");
    txtApellido.setText("");
    txtSapellido.setText("");
    txtEdad.setText("");
}

void limpiarTabla(DefaultTableModel model) {
    for (int i = 0; i <= TablaDatos.getRowCount(); i++) {
        model.removeRow(i);
        i = i - 1;
    }
}

void Actualizar() {
    limpiarTabla(model);
    listar();
}
```

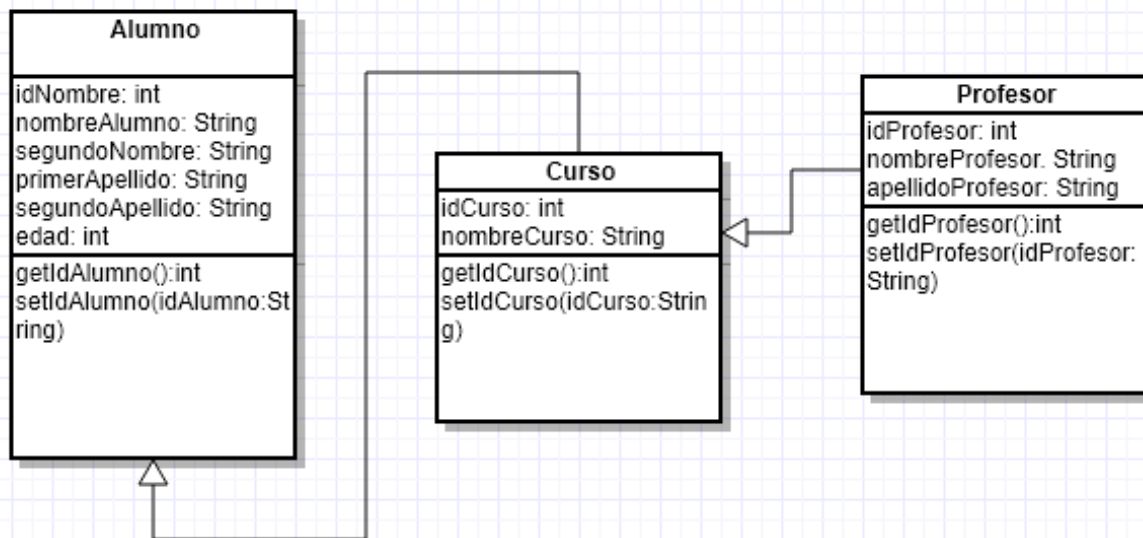
1. **NetBeans:** es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE. Es un producto libre y gratuito sin restricciones de uso.



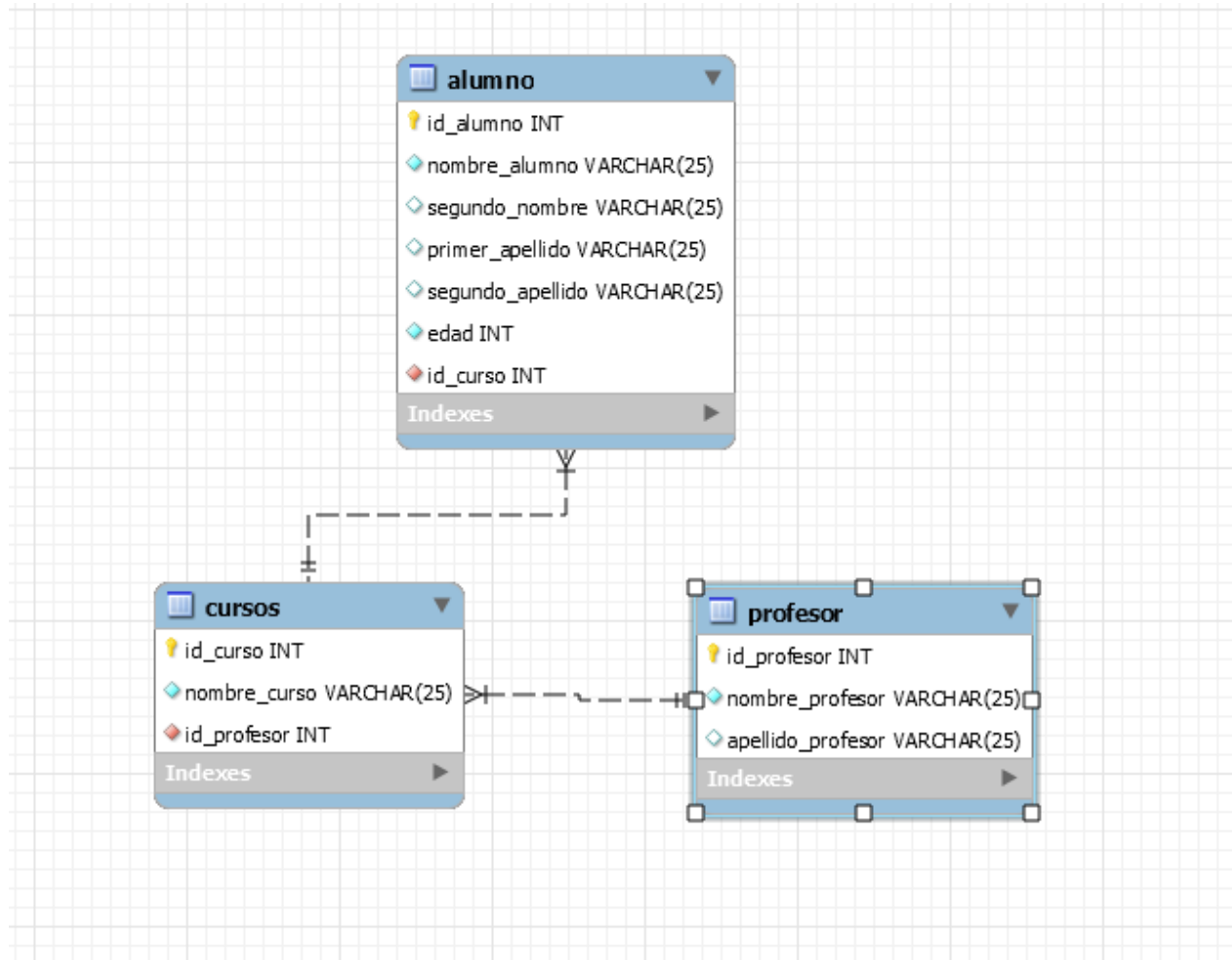
2. **MySQL:** es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, todo para entornos de desarrollo web.



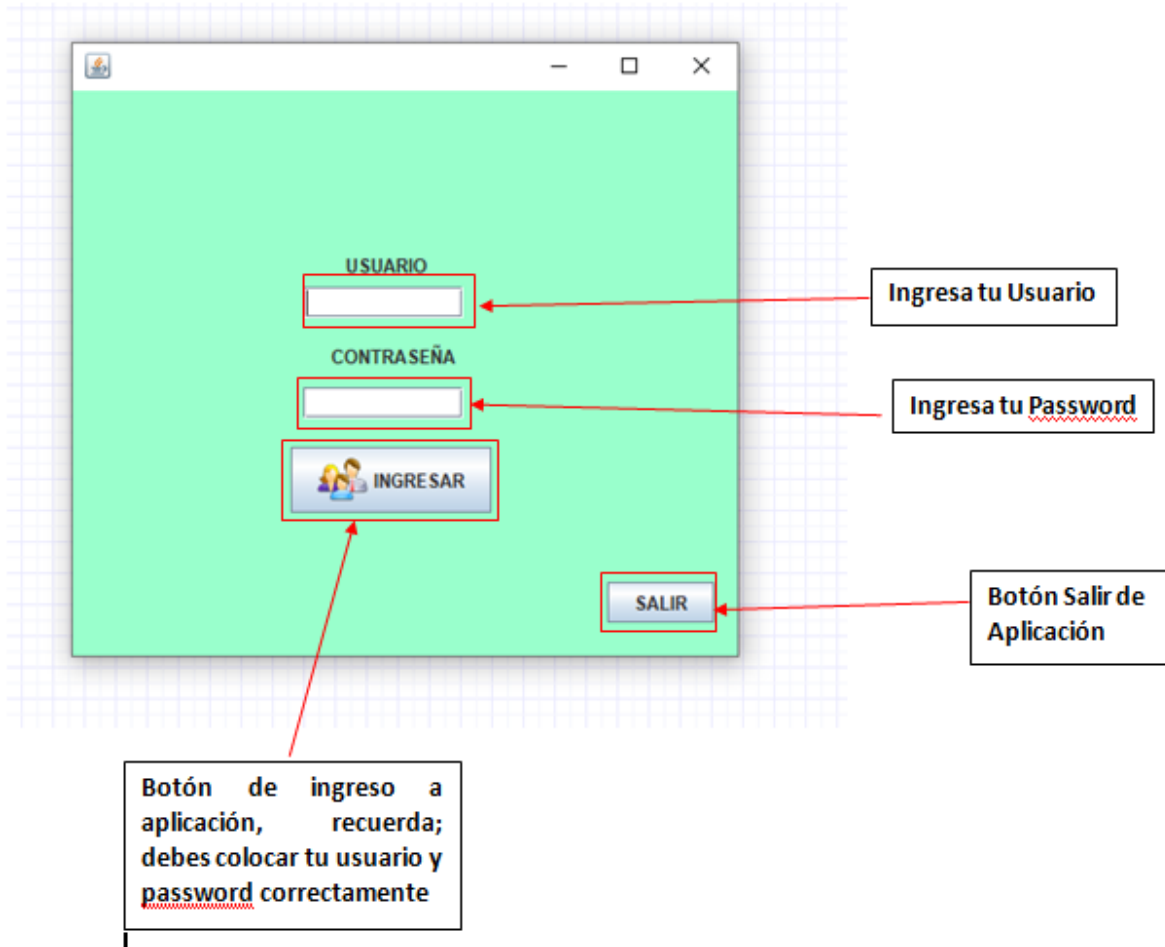
DIAGRAMA DE CLASES



ER FÍSICO



MANUAL DE USUARIO



Datos a llenar para registrar alumno

The screenshot shows a web application window titled "REGISTRO". It contains several sections:

- Datos:** A section with input fields for "ID:", "NOMBRE", "SEGUNDO NOMBRE", "PRIMER APELLIDO", "SEGUNDO APELLIDO", and "EDAD".
- Operaciones:** A section with four buttons: "Agre..." (Add), "Modifi..." (Modify), "Eliminar" (Delete), and "Nue..." (New).
- REGRESAR:** A button labeled "REGRESAR" with a circular arrow icon.
- Lista:** A table with columns: "ID", "PRIMER_NOMBRE", "SEGUNDO_NOMBRE", "PRIMER_APELLIDO", "SEGUNDO_APELLIDO", and "EDAD". The table is currently empty.

Red arrows point from external text boxes to specific elements: one to the "Datos" section, one to the "Operaciones" buttons, one to the "REGRESAR" button, and one to the "Lista" table.

Botones disponibles para realizar tus Gestiones:
Guardar, Modificar, Eliminar, Nuevo

Botón para regresar a Logín.

Desplegado de Datos
Ingresados y Almacenados.

CONCLUSIONES

- MySQL es un sistema de gestión de base de datos relacional, basado en lenguaje de consulta estructurado, se puede ejecutar en prácticamente todas las plataformas
- La plataforma de NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos.
- La POO es una forma especial de programar, este paradigma es cercano a como expresamos las cosas en la vida real en nuestro día a día.