

“AÑO DE LA UNIDAD, LA PAZ Y EL DESARROLLO”

UNIVERSIDAD PERUANA LOS ANDES

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE SISTEMAS Y COMPUTACIÓN



PROYECTOS DEL CURSO DE ARQUITECTURA DE SOFTWARE

Integrantes:

- DIEGO PAOLO BRAVO SORA

Materia: ARQUITECTURA DE SOFTWARE

Sección: B1

Ciclo: VII

2024

Huancayo-Perú

PROYECTO 03

Enunciado del proyecto

Título del proyecto: Sistema de Registro de Trabajadores

Descripción:

La aplicación permite registrar, eliminar y visualizar trabajadores en el sistema, facilitando la administración del personal. Los usuarios pueden ingresar los detalles de los trabajadores, como nombre, puesto y salario, para llevar un control del personal.

Requerimientos funcionales

Registrar trabajador:

Los usuarios pueden agregar nuevos trabajadores al sistema, proporcionando atributos como nombre, puesto y salario.

Código:	<input type="text"/>	
Nombres:	<input type="text"/>	<input type="button" value="NUEVO"/>
Apellidos:	<input type="text"/>	<input type="button" value="REGISTRAR"/>
Fecha de nacimiento:	<input type="text"/>	<input type="button" value="MODIFICAR"/>
Fecha de contratación:	<input type="text"/>	<input type="button" value="ELIMINAR"/>

Eliminar trabajador:

Los usuarios pueden eliminar los trabajadores registrados en el sistema.

<input type="button" value="NUEVO"/>
<input type="button" value="REGISTRAR"/>
<input type="button" value="MODIFICAR"/>
<input type="button" value="ELIMINAR"/>

Ver lista de trabajadores:

Los usuarios pueden visualizar todos los trabajadores registrados.

Editar información del trabajador (opcional):

Los usuarios pueden modificar los detalles de un trabajador existente.

Código:

Nombres:

Apellidos:

Fecha de nacimiento:

Fecha de contratación:

NUEVO

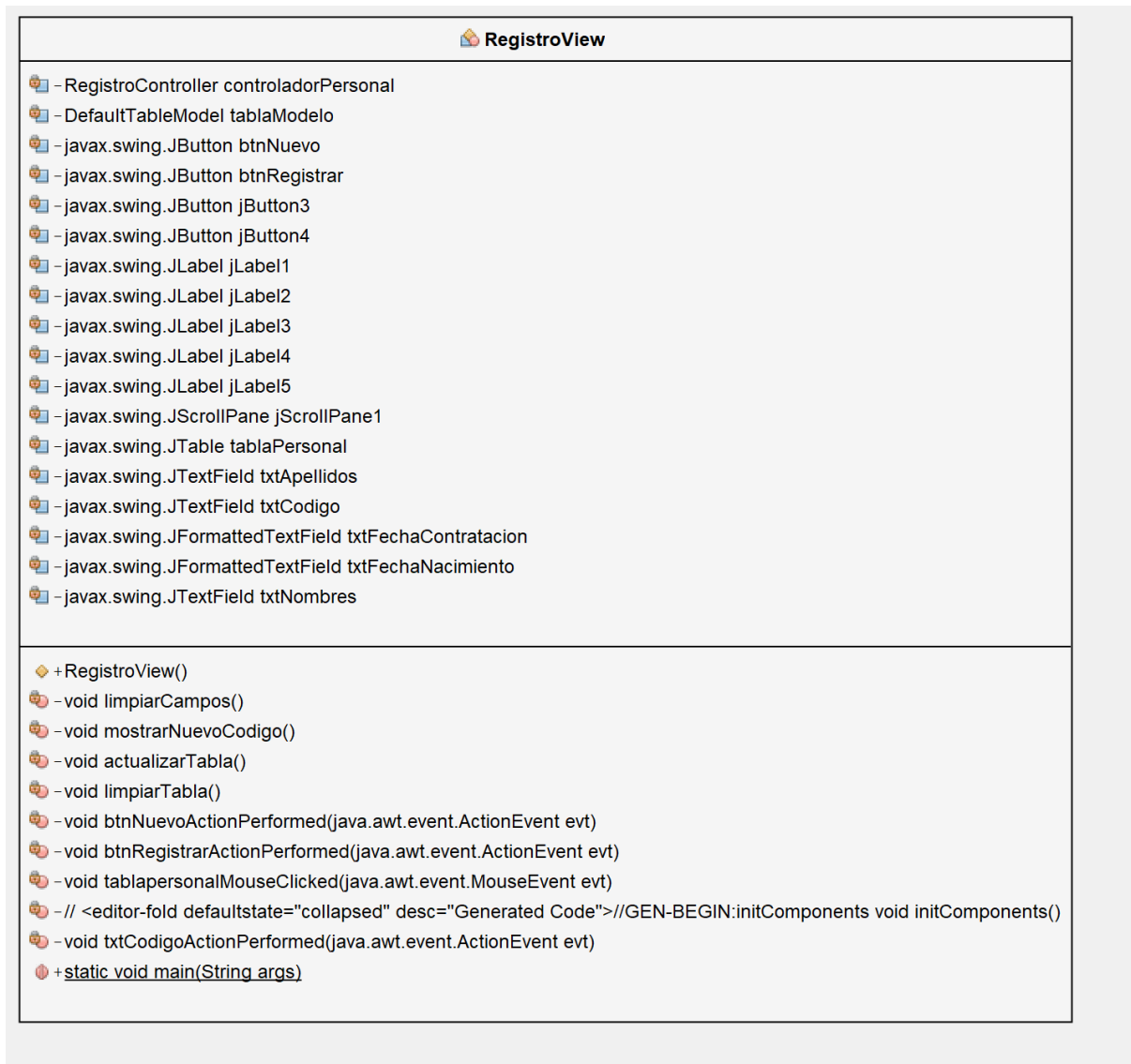
REGISTRAR

MODIFICAR

ELIMINAR

Código	Nombres	Apellidos	Fecha de nacimiento	Fecha de contratación

Diagrama de clases



Código de la aplicación

```
import Controller.RegistroController;
import Modelo.Personal;
import Modelo.RegistroPersonal;
import java.time.format.DateTimeFormatter;
import java.util.ArrayList;
import javax.swing.JOptionPane;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;

/**
 *
 * @author USER 17
 */
public class RegistroView extends javax.swing.JFrame {
    private RegistroController controladorPersonal;
    private DefaultTableModel tablaModelo;

    public RegistroView() {
        initComponents();
        controladorPersonal = new RegistroController();
        tablaModelo = (DefaultTableModel) tablaPersonal.getModel();

        // Configurar eventos para los botones y la tabla
        btnNuevo.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnNuevoActionPerformed(evt);
            }
        });

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

        tablaPersonal.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                tablapersonalMouseClicked(evt);
            }
        });

        // Inicializar tabla con datos iniciales si es necesario
```

```

        actualizarTabla();
    }

    // Método para limpiar los campos de entrada
    private void limpiarCampos() {
        txtNombres.setText("");
        txtApellidos.setText("");
        txtFechaNacimiento.setText("");
        txtFechaContratacion.setText("");
    }

    // Método para mostrar el nuevo código de personal
    private void mostrarNuevoCodigo() {
        int siguienteCodigo = controladorPersonal.obtenerListaPersonal().size() + 1;
        String nuevoCodigo = "P" + String.format("%02d", siguienteCodigo);
        txtCodigo.setText(nuevoCodigo);
    }

    // Método para actualizar la tabla con los datos del controladorPersonal
    private void actualizarTabla() {
        limpiarTabla();
        ArrayList<Personal> lista = controladorPersonal.obtenerListaPersonal();
        for (Personal personal : lista) {
            Object[] fila = {personal.getCodigo(), personal.getNombres(), personal.getApellidos(),
personal.getFechaNacimiento(), personal.getFechaContratacion()};
            tablaModelo.addRow(fila);
        }
    }

    // Método para limpiar la tabla
    private void limpiarTabla() {
        int filas = tablaModelo.getRowCount();
        for (int i = filas - 1; i >= 0; i--) {
            tablaModelo.removeRow(i);
        }
    }

    // Evento del botón Nuevo
    private void btnNuevoActionPerformed(java.awt.event.ActionEvent evt) {
        limpiarCampos();
        mostrarNuevoCodigo();
    }

    // Evento del botón Registrar
    private void btnRegistrarActionPerformed(java.awt.event.ActionEvent evt) {
        String nombres = txtNombres.getText();
        String apellidos = txtApellidos.getText();
        String fechaNacimiento = txtFechaNacimiento.getText();
    }

```

```

        String fechaContratacion = txtFechaContratacion.getText();

        controladorPersonal.agregarPersonal(nombres, apellidos, fechaNacimiento,
fechaContratacion);
        actualizarTabla();
        limpiarCampos();
        mostrarNuevoCodigo();
    }

    // Evento del clic en la tabla
    private void tablapersonalMouseClicked(java.awt.event.MouseEvent evt) {
        int indiceFilaSeleccionada = tablaPersonal.getSelectedRow();
        if (indiceFilaSeleccionada != -1) {
            Personal personalSeleccionado =
controladorPersonal.obtenerListaPersonal().get(indiceFilaSeleccionada);
            txtNombres.setText(personalSeleccionado.getNombres());
            txtApellidos.setText(personalSeleccionado.getApellidos());
            txtFechaNacimiento.setText(personalSeleccionado.getFechaNacimiento());
            txtFechaContratacion.setText(personalSeleccionado.getFechaContratacion());
        }
    }
}

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

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    txtNombres = new javax.swing.JTextField();
    txtApellidos = new javax.swing.JTextField();
    jScrollPane1 = new javax.swing.JScrollPane();
    tablaPersonal = new javax.swing.JTable();
    btnNuevo = new javax.swing.JButton();
    btnRegistrar = new javax.swing.JButton();
    jButton3 = new javax.swing.JButton();
    jButton4 = new javax.swing.JButton();
    txtFechaNacimiento = new javax.swing.JFormattedTextField();
    txtFechaContratacion = new javax.swing.JFormattedTextField();
    jLabel5 = new javax.swing.JLabel();

```



```

txtCodigo = new javax.swing.JTextField();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setBackground(new java.awt.Color(255, 204, 102));

jLabel1.setForeground(new java.awt.Color(255, 153, 51));
jLabel1.setText("Nombres:");

jLabel2.setForeground(new java.awt.Color(255, 153, 51));
jLabel2.setText("Apellidos:");

jLabel3.setForeground(new java.awt.Color(255, 153, 51));
jLabel3.setText("Fecha de nacimiento:");

jLabel4.setForeground(new java.awt.Color(255, 153, 51));
jLabel4.setText("Fecha de contratación:");

txtNombres.setForeground(new java.awt.Color(255, 153, 51));

txtApellidos.setForeground(new java.awt.Color(255, 153, 51));

jScrollPane1.setForeground(new java.awt.Color(255, 153, 51));

tablaPersonal.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "Código", "Nombres", "Apellidos", "Fecha de nacimiento", "Fecha de contratación"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, true, true
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});
jScrollPane1.setViewportViewView(tablaPersonal);

btnNuevo.setForeground(new java.awt.Color(255, 153, 51));
btnNuevo.setText("NUEVO");

btnRegistrar.setForeground(new java.awt.Color(255, 153, 51));

```

```

btnRegistrar.setText("REGISTRAR");

jButton3.setForeground(new java.awt.Color(255, 153, 51));
jButton3.setText("MODIFICAR");

jButton4.setForeground(new java.awt.Color(255, 153, 51));
jButton4.setText("ELIMINAR");

txtFechaNacimiento.setForeground(new java.awt.Color(255, 153, 51));

txtFechaContratacion.setForeground(new java.awt.Color(255, 153, 51));

jLabel5.setForeground(new java.awt.Color(255, 153, 51));
jLabel5.setText("Código:");

txtCodigo.setForeground(new java.awt.Color(255, 153, 51));
txtCodigo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtCodigoActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(40, 40, 40)
            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 786,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(51, Short.MAX_VALUE))
        .addGroup(layout.createSequentialGroup()
            .addGap(58, 58, 58)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel1)
            .addComponent(jLabel2)
            .addComponent(jLabel3)
            .addComponent(jLabel4)
            .addComponent(jLabel5)
            .addGap(115, 115, 115)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(txtNombres)
            .addComponent(txtApellidos)
            .addComponent(txtFechaNacimiento,
javax.swing.GroupLayout.DEFAULT_SIZE, 218, Short.MAX_VALUE)
            .addComponent(txtFechaContratacion)

```

```

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

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(jButton3, javax.swing.GroupLayout.DEFAULT_SIZE, 106,
Short.MAX_VALUE)
            .addComponent(btnRegistrar, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(btnNuevo, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addGroup(100, 100, 100))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addComponent(jLabel5)
                .addComponent(txtCodigo, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel1)
                .addComponent(txtNombres, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(btnNuevo))
            .addGroup(18, 18, 18)

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel2)
                .addComponent(txtApellidos, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(btnRegistrar))
            .addGroup(18, 18, 18)

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel3)
                .addComponent(jButton3)
                .addComponent(txtFechaNacimiento,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(18, 18, 18)

```

```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel4)
    .addComponent(jButton4)
    .addComponent(txtFechaContratacion,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(40, 40, 40)
    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addContainerGap(77, Short.MAX_VALUE))
);

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

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

public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and
feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException | InstantiationException | IllegalAccessException |
javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(RegistroView.class.getName()).log(java.util.logging.Level.
SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(() -> {

```

```
        new RegistroView().setVisible(true);
    });
}

// Variables declaration - do not modify
private javax.swing.JButton btnNuevo;
private javax.swing.JButton btnRegistrar;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable tablaPersonal;
private javax.swing.JTextField txtApellidos;
private javax.swing.JTextField txtCodigo;
private javax.swing.JFormattedTextField txtFechaContratacion;
private javax.swing.JFormattedTextField txtFechaNacimiento;
private javax.swing.JTextField txtNombres;
// End of variables declaration
}
```