



# ESCUELA POLITÉCNICA NACIONAL

## ESCUELA DE FORMACIÓN DE TECNÓLOGOS



### PROGRAMACION (TDSD314)

ASIGNATURA:

PROGRAMACION ORIENTADA A OBJETOS

PROFESOR:

Ing. Juan Carlos Gonzalez MSc.

PERÍODO ACADÉMICO:

2025-B

## Deber N° 10

### TÍTULO

## TALLER DE PROGRAMACIÓN ORIENTADA A OBJETOS

ESTUDIANTE:

<<Edwin Efraín Chusin Enríquez>>

### OBJETIVO GENERAL

Comprender los conceptos básicos de programación orientada a objetos

### OBJETIVOS ESPECÍFICOS

- Realizar el formulario solicitado

## DESARROLLO

El desarrollo del presente taller consiste en realizar lo siguiente:

### CODIFICACIÓN DE LA APLICACION

Diseñar una interfaz gráfica en Java Swing que permita ingresar los datos de una persona simulando una hoja de vida y guardar en un archivo .txt (consultar ficheros en java)

Código de las clases usadas

#### Clase Persona

```
package Entidad;

public class Persona {
    private String nombres;
    private String apellidos;
    private String fechaNacimiento;
    private String ci;
    private String telefono;
    private String correo;
    private String direccion;
    private String puesto;
    private String discapacidad;
    private String detalle;
    private String idiomas;
    private String nivel;
    private String habilidades;

    public Persona(String nombres, String apellidos, String
fechaNacimiento, String ci, String telefono, String correo, String
direccion, String puesto, String discapacidad, String detalle, String
idiomas, String nivel, String habilidades) {
        this.nombres = nombres;
        this.apellidos = apellidos;
        this.fechaNacimiento = fechaNacimiento;
        this.ci = ci;
        this.telefono = telefono;
        this.correo = correo;
        this.direccion = direccion;
        this.puesto = puesto;
        this.discapacidad = discapacidad;
        this.detalle = detalle;
        this.idiomas = idiomas;
        this.nivel = nivel;
        this.habilidades = habilidades;
    }

    public int calcularEdad() {
        String[] fecha = fechaNacimiento.split("-");
        int anioNacimiento = Integer.parseInt(fecha[0]);
        return 2025 - anioNacimiento;
    }

    @Override
    public String toString() {
```

```

        return "\n===== "+
               "\n                      HOJA DE VIDA                      "+
               "\n===== "+
               "\nNombres: "+nombres+ "                      Apellidos: "+apellidos+
               "\nCedula de identidad: "+ci+ "                      " +
               "\nFecha de nacimiento: "+fechaNacimiento+
               "\nEdad: "+calcularEdad()+" años"+
               "\n"+
               "\nCorreo electrónico: "+correo+
               "\nTeléfono: "+telefono+ "                      Dirección:
"+direccion+
               "\nPuesto de trabajo: "+puesto+
               "\n"+
               "\nDiscapacidad: "+discapacidad+
               "\nDetalle de discapacidad: "+detalle+
               "\n"+
               "\nIdiomas: "+idiomas+ "                      Nivel: "+nivel+
               "\nHabilidades: "+habilidades;
    }
}

```

### Clase HojaDeVida que extiende de JFrame

```

package Entidad;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.FileWriter;
import java.io.IOException;

public class HojaDeVida extends JFrame{
    private JLabel lblTitulo;
    private JLabel lblNombre;
    private JLabel lblApellido;
    private JLabel lblPuestoTrabajo;
    private JTextField txtNombres;
    private JTextField txtApellidos;
    private JTextField txtTrabajo;
    private JTextField txtEmail;
    private JTextField txtCelular;
    private JTextField txtLocalidad;
    private JLabel lblCorreo;
    private JLabel lblTelefono;
    private JLabel lblDireccion;
    private JTextField txtFechaNacimiento;
    private JLabel lblNacimiento;
    private JRadioButton rdbSi;
    private JLabel lblDetalles;
    private JTextField txtDetalleDiscapacidad;
    private JLabel lblHabilidades;
    private JCheckBox chkComunicacion;
    private JCheckBox chkLiderazgo;
    private JCheckBox chkTomaDecision;
    private JCheckBox chkTrabajaEnEquipo;
    private JLabel lblIdiomas;
    private JTextField txtIdiomas;
    private JComboBox cbNivel;
}

```

```

private JLabel lblNivel;
private JButton btnGuardar;
private JButton btnLimpiar;
private JPanel Panel1;
private JRadioButton rdbNo;
private JTextField txtCedula;
private JLabel lblCedula;
private JLabel lblTieneDiscapacidad;
private ButtonGroup grupoDiscapacidad;

public HojaDeVida() throws HeadlessException {
    setContentPane(Panel1);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setSize(700, 600);
    setLocationRelativeTo(null);

    Panel1.setBorder(
        BorderFactory.createEmptyBorder(20, 40, 30, 0) //Inge es
para dejar un espacio vacio en los bordes,
    );
//esq se ve feo si no esta centrado.
    grupoDiscapacidad = new ButtonGroup();
    grupoDiscapacidad.add(rdbSi);
    grupoDiscapacidad.add(rdbNo);

    btnGuardar.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            guardarDocumento();
        }
    });

    btnLimpiar.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            limpiar();
        }
    });
}

public void guardarDocumento() {
    String Nombre = txtNombres.getText();
    String Apellido = txtApellidos.getText();
    String FechaDeNacimiento = txtFechaNacimiento.getText();
    String cedula = txtCedula.getText();
    String PuestoDeTrabajo = txtTrabajo.getText();
    String Telefono = txtCelular.getText();
    String Correo = txtEmail.getText();
    String Direccion = txtLocalidad.getText();
    String Idiomas = txtIdiomas.getText();
    String nivel = cbNivel.getSelectedItem().toString();
    String discapacidad = rdbSi.isSelected()?"Si":"No";
    String detalle = txtDetalleDiscapacidad.getText();

    String habilidades = "";
    if (chkLiderazgo.isSelected()) habilidades += "\n    -
Liderazgo";
    if (chkComunicacion.isSelected()) habilidades += "\n    -
Comunicación efectiva";
    if (chkTomaDecision.isSelected()) habilidades += "\n    - Toma de
decisiones";
    if (chkTrabajaEnEquipo.isSelected()) habilidades += "\n    -
Trabajo en equipo";

```

```
        Persona p1 = new Persona(Nombre, Apellido, FechaDeNacimiento,
cedula, Telefono, Correo, Direccion,
PuestoDeTrabajo, discapacidad, detalle, Idiomas, nivel, habilidades);

        long tiempo = System.currentTimeMillis();
        String nombreArchivo = "Hoja_vida_"+tiempo+".txt";
        try{
            FileWriter nuevoArchivo = new FileWriter(nombreArchivo, true);
            nuevoArchivo.write(p1.toString());
            nuevoArchivo.close();

            JOptionPane.showMessageDialog(this, "Datos guardados
correctamente");

        }catch (IOException ex){
            JOptionPane.showMessageDialog(this, "Error al guardar el
archivo");
        }
    }

    public void limpiar(){
        txtNombres.setText("");
        txtApellidos.setText("");
        txtCedula.setText("");
        txtIdiomas.setText("");
        txtCelular.setText("");
        txtDetalleDiscapacidad.setText("");
        txtEmail.setText("");
        txtFechaNacimiento.setText("YYYY-MM-DD");
        txtLocalidad.setText("");
        txtTrabajo.setText("");

        rdbSi.setSelected(false);
        rdbNo.setSelected(false);

        cbNivel.setSelectedIndex(0);

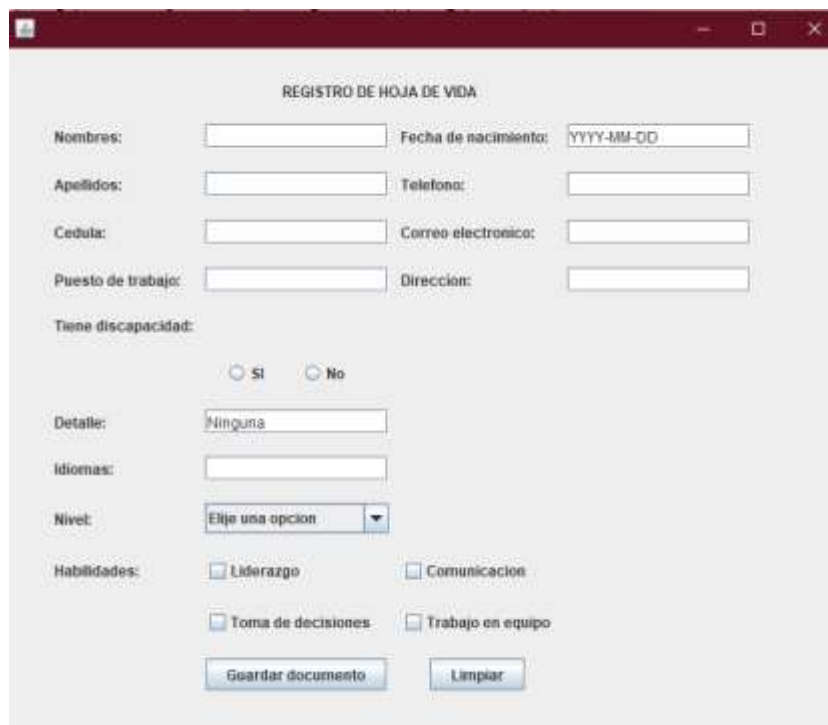
        chkLiderazgo.setSelected(false);
        chkComunicacion.setSelected(false);
        chkTomaDecision.setSelected(false);
        chkTrabajaEnEquipo.setSelected(false);
    }

    public static void main(String[] args) {
        HojaDeVida h1 = new HojaDeVida();
        h1.setVisible(true);
    }
}
```

## EJECUCIÓN

Agregue capturas de pantalla indicando la funcionalidad de la aplicación y demostrar que la información se guarda en el archivo.

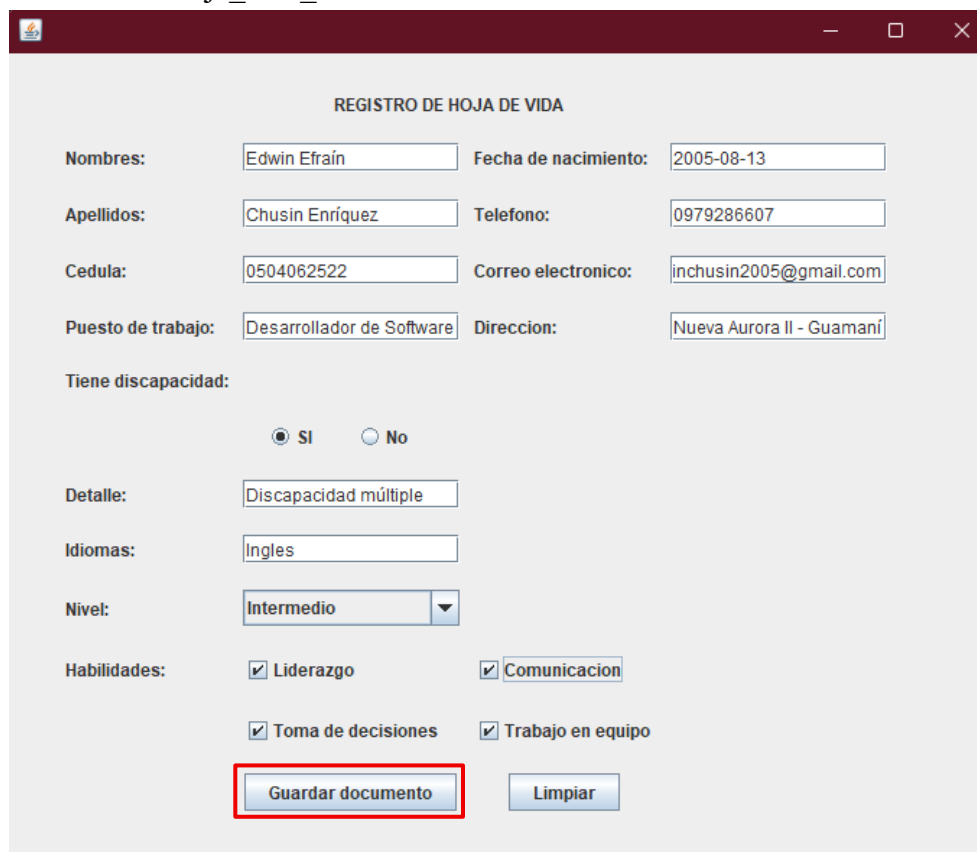
Al iniciar la aplicación.



The screenshot shows a web application window titled "REGISTRO DE HOJA DE VIDA". The form contains the following fields and controls:

- Nombres:
- Fecha de nacimiento:
- Apellidos:
- Telefono:
- Cedula:
- Correo electronico:
- Puesto de trabajo:
- Direccion:
- Tiene discapacidad:   
☐ Si ☐ No
- Detalle:
- Idiomas:
- Nivel:
- Habilidades:   
☐ Liderazgo ☐ Comunicacion   
☐ Toma de decisiones ☐ Trabajo en equipo
- Buttons: "Guardar documento" and "Limpiar"

Después de llenar el formulario, le damos click a guardar documento y se guarda en formato.txt de nombre “Hoja\_vida\_numeroRandom.txt”.



The screenshot shows the same web application window, but now the form is filled with data. The "Guardar documento" button is highlighted with a red rectangle.

REGISTRO DE HOJA DE VIDA

Nombres:  Fecha de nacimiento:

Apellidos:  Telefono:

Cedula:  Correo electronico:

Puesto de trabajo:  Direccion:

Tiene discapacidad:   
☒ Si ☐ No

Detalle:

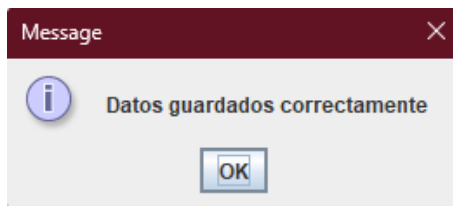
Idiomas:

Nivel:

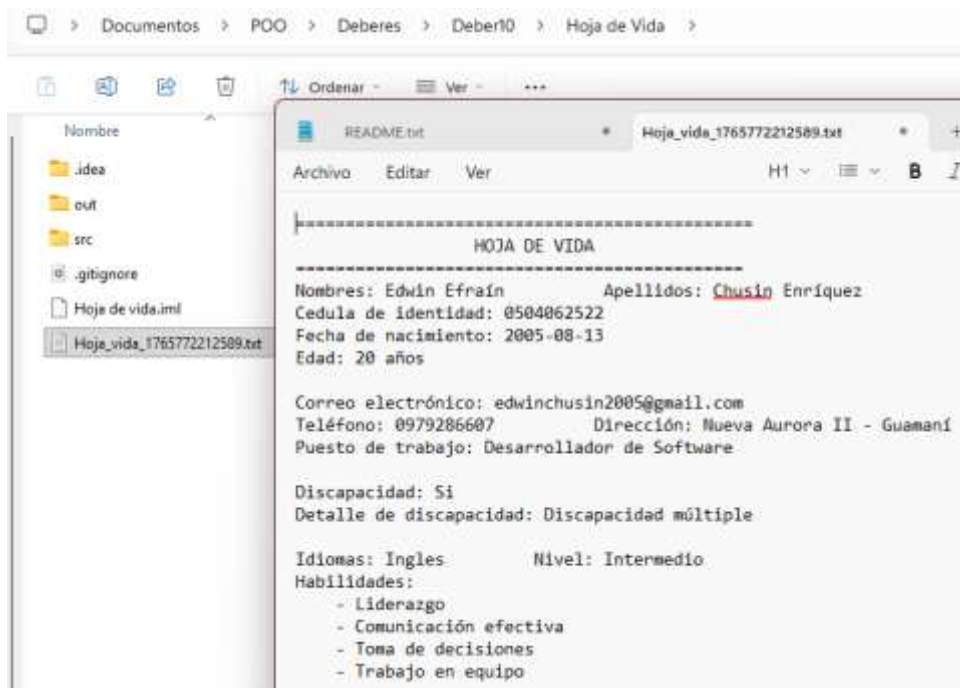
Habilidades:   
☒ Liderazgo ☒ Comunicacion   
☒ Toma de decisiones ☒ Trabajo en equipo

Buttons: "Guardar documento" (highlighted) and "Limpiar"

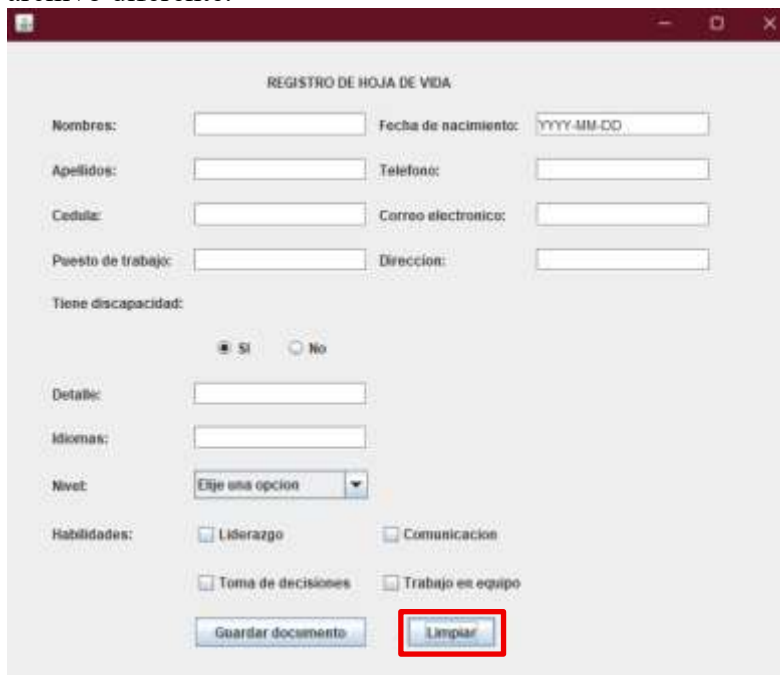
Solo como ejemplo ponemos discapacidad múltiple, una vez se de click en guardar el programa notificara si se guardó correctamente o no.



Y el archivo se creará en la misma carpeta donde se encuentra el programa.



Para generar otro archivo se da a limpiar y sigue el mismo procedimiento, lo que generara otro archivo diferente.

A screenshot of a web form titled "REGISTRO DE HOJA DE VIDA". The form contains various input fields for personal and professional information, including names, dates, phone numbers, email addresses, and addresses. It also has radio buttons for "Tiene discapacidad" (Yes/No) and a dropdown for "Nivel". At the bottom, there are checkboxes for "Habilidades" (Leadership, Communication, Decision Making, Teamwork) and two buttons: "Guardar documento" and "Limpiar". The "Limpiar" button is highlighted with a red rectangle.

## **REPOSITORIO**

Link de Github: [https://github.com/EdwinC-h/Deber10\\_Edwin\\_Chusin.git](https://github.com/EdwinC-h/Deber10_Edwin_Chusin.git)