

Manual Técnico

El proyecto fue realizado en su totalidad utilizando el lenguaje de programación Java en su versión 19.0.2 en adición de librerías para manejar documentos .csv, .pdf y .bin

```
// @author fernandoorozco
package pantallas;
import clases.Administrador;
import clases.Profesor;
import clases.Alumno;
import clases.Cursor;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.util.ArrayList;
import javax.swing.JOptionPane;

public class Login extends javax.swing.JFrame {
    Administrador admin = new Administrador( usuario:"a", password:"a");
    Profesor profesor = new Profesor( usuario:"b", password:"b");
    Alumno alumno = new Alumno( usuario:"c", password:"c");

    moduloAdmin pantallaAdministrador = new moduloAdmin();
    moduloProfesores pantallaProfesor;
    moduloAlumnos pantallaEstudiante;

    public static int codigoUsuarioActualProfesor;
    public static int indexActualProfesor;
    public static int codigoUsuarioActualAlumno;
    public static int indexActualAlumno;
    public static String nombreUsuarioActual;
    public static String nombreProfesorActual;

    public Login() {
        initComponents();
        this.recuperarProfesores();
        this.recuperarAlumnos();
        this.recuperarCursos();
    }

    @SuppressWarnings("unchecked")
    Generated Code

    private void entrarCuentaButtonActionPerformed(java.awt.event.ActionEvent evt) {...36 lines }

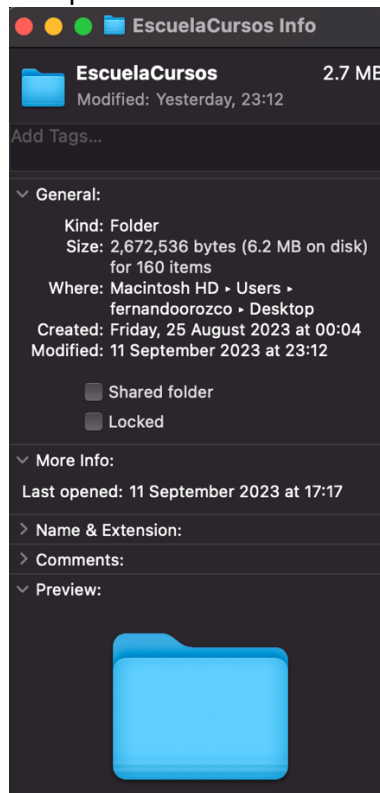
    public boolean buscarUsuarioProfesor(String userString, String password){...15 lines }

    public boolean burcarUsuarioAlumno(String userString, String password){...15 lines }

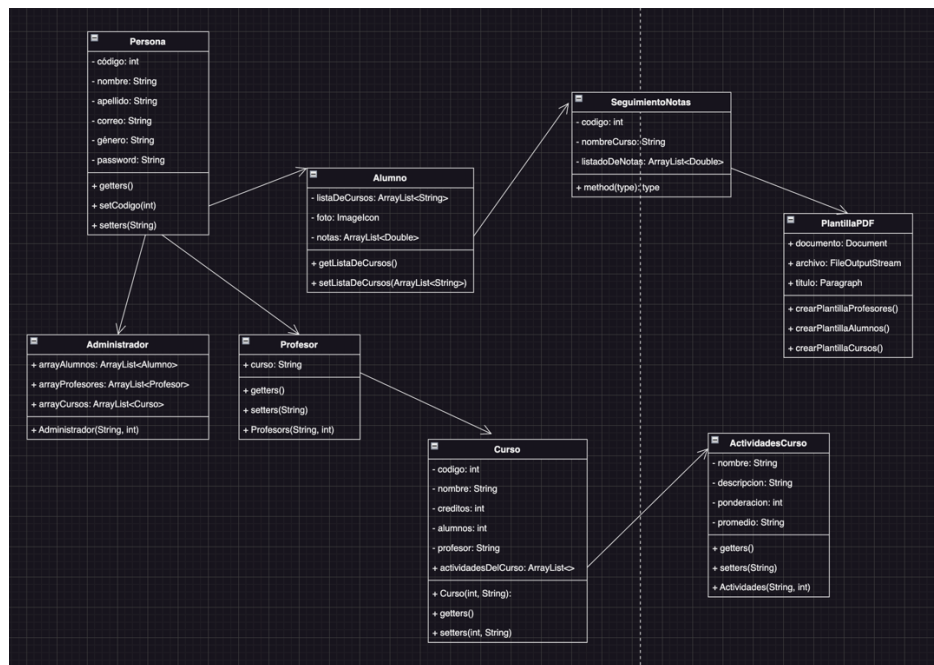
    // Funciones para recuperar Datos, Persistencia de Datos
    public void recuperarProfesores() {...24 lines }

    public void recuperarAlumnos() {...24 lines }
```

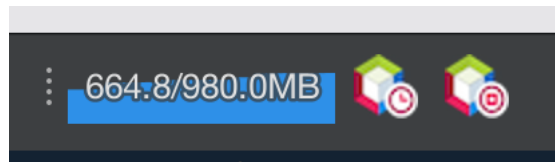
El código fuente del proyecto tiene un peso aproximado de 2.7MB



Se realizó un diagrama de clases que representa el flujo de las clases y sus métodos que dan funcionalidad a toda la aplicación, así como las clases que se encargan de la interfaz que esta utiliza.



Al ejecutarse el programa la memoria que ocupa es de menos de 664.8 MB, existe aspectos para poder optimizarlo con el fin de reducir estos recursos



El programa se desarrolló con el uso de una interfaz gráfica a diferencia de la primera práctica la cual solo uso la consola como medio de interacción con el usuario. Desde un simple Log-in hasta manejo de tablas

A login window titled 'DTT'. It features two input fields: 'usuario' and 'contraseña'. Below these fields is a button labeled 'Iniciar'.A dashboard titled 'Python' with a 'Refrescar Datos de Tablas' button. It is divided into several sections:

- Listado Alumnos:** A table with columns 'Código', 'Nombre', 'Apellido', and 'Acciones'. It lists 12 students with their IDs and names.
- Actividades:** A table with columns 'Nombre', 'Descripción', 'Ponderación', and 'Promedio'. It lists three activities: 'Tarea 1', 'Parcial 1', and 'Examen Final'.
- Acumulado:** A summary row showing 'Acumulado: 80 /100'.
- Reportes:** Two buttons: 'Top 5 - Estudiantes con MEJOR rendimiento' and 'Top 5 - Estudiantes con PEOR rendimiento'.
- Crear Actividad:** A form with fields for 'Nombre', 'Descripción', and 'Ponderación', and a 'Selecciónar archivo CSV' button.
- Buttons:** 'CERRAR' and 'Crear Actividad'.

Se implementaron ciertas librerías para el desarrollo de la aplicación, como por ejemplo las librerías awt, ArrayList, swing y io.FileInputStream

```
// @author fernandoorozco
package pantallas;
import clases.Administrador;
import clases.Profesor;
import clases.Alumno;
import clasesCurso;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.util.ArrayList;
import javax.swing.JOptionPane;
```

Se implementaron los conceptos de modularidad y el paradigma de programación de “Programación Orientada a Objetos”

