



ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS



PROGRAMACIÓN ORIENTADAS A OBJETOS

(TDSD314)

GR1

PROFESOR: YADIRA FRANCO

FECHA: 7 de agosto de 2025

PERÍODO ACADÉMICO: 2025-A

RECUPERACIÓN

DE PROGRAMACIÓN ORIENTADA A OBJETOS

Integrantes:

Carlos Serrano
Kevin Simbaña
Henry Tonato
Felipe Zapata

Link de Git:

Índice

Contenido

- 1. Título del Proyecto**
- 2. Objetivos**
- 3. Justificación**
- 4. Alcance y Requerimientos**
- 5. Requerimientos**
- 6. Requerimientos Tecnológicos**
- 7. Ejecución Completa del Proyecto**
- 8. Explicación del Desarrollo**

1. Título del Proyecto

Sistema CRUD

2. Objetivos

a. Objetivo General

Desarrollar un sistema que permita realizar las funciones principales de un CRUD (Crear, leer, actualizar y eliminar).

b. Objetivos Específicos

- Poner en práctica los conceptos de Programación orientada a objetos en un proyecto.
- Estructurar formularios usando la herramienta Java Swing.
- Conectar una base de datos MySQL Clever Cloud para almacenar la información.

3. Justificación

La realización de este proyecto tiene el fin de demostrar nuestros conocimientos en el desarrollo de un sistema CRUD junto a la correcta conexión a una base de datos MySQL en Clever Cloud y con esto aprobar la materia de Programación Orientada a Objetos.

4. Alcance y Requerimientos

a. Requerimientos

- El sistema debe permitir el login de usuarios.
- El usuario admin pueda acceder a la tabla de usuarios del sistema.
- El usuario admin pueda registrar nuevos usuarios.
- El usuario admin pueda actualizar la información de un usuario seleccionado.
- El usuario admin pueda eliminar la información de un usuario.

b. Requerimientos Tecnológicos

- **Java JDK 21**

El proyecto fue desarrollado usando Java JDK 24, compatible con la sintaxis moderna de Java y nuevas características del lenguaje.

- **IDE IntelliJ IDEA**

Todo el desarrollo se hizo en IntelliJ IDEA, aprovechando la herramienta de diseño GUI JavaSwing.

- **Base de Datos MySQL en nube**

El sistema está conectado a una base de datos MySQL alojada en Clever Cloud, accesible remotamente a través del jdbc:mysql://... con las credenciales del Clever Cloud.

- **Librerías JDBC**

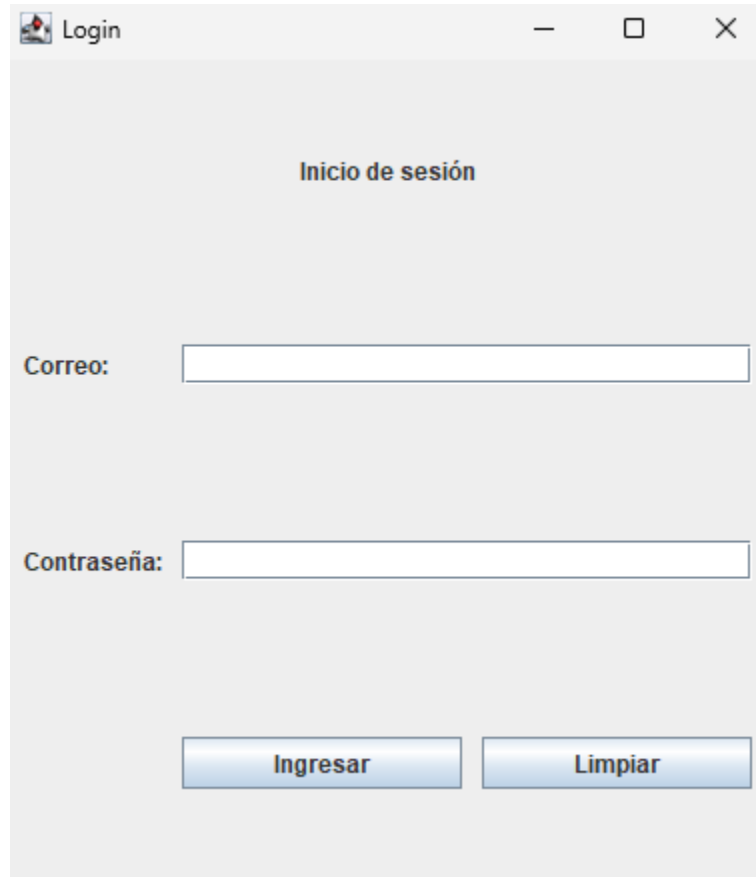
El acceso a la base de datos se hace mediante JDBC (mysql-connector-j-9.4.0), utilizando java.sql.Connection, PreparedStatement, ResultSet, etc., con manejo de errores y consultas SQL

- **GitHub para control de versiones**

El proyecto se encuentra alojado en un repositorio en GitHub, con los commits correspondientes que muestran la evolución y avance del proyecto.

5. Ejecución Completa del Proyecto

LOGIN:




A screenshot of a Java Swing window titled "Login". The window has a standard title bar with minimize, maximize, and close buttons. The main content area has a light gray background. At the top center, the text "Inicio de sesión" is displayed. Below this, there are two input fields. The first is labeled "Correo:" and the second is labeled "Contraseña:". At the bottom, there are two buttons: "Ingresar" and "Limpiar".

Inicio de sesión

Correo:

Contraseña:

 Login

— □ ×

Inicio de sesión


Correo:

Contraseña:

.....

Message

×


 Usuario ingresó correctamente

OK

Ingresar

Limpiar

Tabla de usuarios:

 Principal

—

□

×

Tabla de Usuarios

1	elpepe	elpepe@gmail.com
2	carlos	carlos@gmail.com

Registrar Usuario

Actualizar Usuario

Eliminar Usuario

Salir

Registro de usuarios:

Registrar


Registrar Usuario

ID:

Usuario:

Correo:

Message

 Usuario registrado exitosamente.

Principal

Tabla de Usuarios		
1	elpepe	elpepe@gmail.com
2	carlos	carlos@gmail.com
3	sebastian	sebastian@gmail.com

Actualización de datos:

Actualizar


Actualizar Usuario

ID:

Usuario:

Correo:

Message

 **Usuario actualizado correctamente.**

Principal

Tabla de Usuarios		
1	elpepe	elpepe@gmail.com
2	carlos	carlos@gmail.com
3	sebastian	sebastian@outlook.com

Eliminar usuarios:

Eliminar

Eliminar Usuario

D:

Eliminar Salir

Confirmar

¿Estás seguro de eliminar el usuario?

Yes No

Message

Usuario eliminado.

OK

Principal

Tabla de Usuarios		
1	elpepe	elpepe@gmail.com
2	carlos	carlos@gmail.com

6. Explicación del Desarrollo

El desarrollo del sistema se basó en una arquitectura modular y estructurada, siguiendo los principios de separación de responsabilidades y buenas prácticas de programación orientada a objetos. A continuación, se detallan los principales componentes desarrollados:

a. Vistas: Interfaces gráficas, eventos.

Las interfaces gráficas de usuario fueron implementadas utilizando **Java Swing**, y se organizaron según los distintos roles del sistema. Cada formulario cumple una función específica dentro del sistema:

- Panel para Login: Nos despliega el modulo de inicio de sesión.
- Panel para Menú de Administrador/Operador: Hay un panel diferente para ambos roles, ya que tienen funciones diferentes.
- Panel de vista de componentes: Los componentes como Gestionar Usuarios o Proveedores tienen su propio panel, también hay paneles de reportes, salir, cotizaciones, estado, y todo dependiendo del rol.

b. Conector

Para establecer la conexión con la base de datos MySQL (hospedada en Clever Cloud), se descargó el **conector JDBC de MySQL** (mysql-connector-java-x.x.xx.jar) y se cargó manualmente en la carpeta de **librerías** del proyecto Java. Esta dependencia permite que el sistema interactúe con la base de datos utilizando el protocolo JDBC.

c. Conexión

El sistema se conecta a una base de datos **MySQL alojada en Clever Cloud**. Para lograrlo:

- Se descargó y configuró el **driver JDBC de MySQL**, el cual se añadió al proyecto desde las librerías.
- Se implementó la clase Conexion.java, que contiene los parámetros necesarios para establecer la conexión con el servidor en la nube (host, usuario, contraseña y puerto).
- Se utilizó un bloque try-catch para capturar posibles excepciones durante la conexión y reportar errores en consola si ocurren fallos.

d. Main

El archivo Main.java representa el punto de entrada del sistema. Desde allí se lanza el VistaLogin que permite a los usuarios ingresar al sistema de acuerdo con sus credenciales. Según el tipo de usuario (administrador o operador), se carga el formulario correspondiente.

e. Ejecutable

El proceso para generar el ejecutable del sistema consta de dos etapas:

1. Generación del .jar:

- Desde el entorno de desarrollo (IntelliJ IDEA), se configura la opción "**Artifacts**" para empaquetar el proyecto completo y generar un archivo .jar.

2. Conversión a .exe:

- Se utilizó la herramienta **Launch4j 3.5** para convertir el archivo .jar en un ejecutable .exe.
- En esta herramienta se especifican:
 - La ruta del archivo .jar generado.
 - El ícono del sistema (.ico).
 - La versión mínima de Java requerida para ejecutar el sistema.

- Esto permite una distribución más sencilla del software, especialmente en equipos sin entorno de desarrollo.

7. Funcionamiento con Base de Datos en la Nube

El sistema está conectado a una base de datos remota desplegada en la nube mediante la plataforma Clever Cloud, utilizando el motor de base de datos MySQL. Este enfoque permite acceder a los datos desde cualquier ubicación con conexión a internet, favoreciendo la disponibilidad del sistema.

La gestión y administración de la base de datos se realizó utilizando la herramienta MySQL Workbench, que facilita la conexión remota, ejecución de consultas SQL, visualización de tablas, y monitoreo en tiempo real de los registros.

Configuración de conexión

Para establecer la conexión desde la aplicación Java, se utilizó la biblioteca **JDBC (Java Database Connectivity)**, configurando la URL de conexión.

Los parámetros de acceso (host, puerto, usuario, contraseña y nombre de la base de datos) fueron proporcionados por Clever Cloud tras la creación del servicio MySQL. Estos valores se integraron en la clase de conexión del sistema (ConexionBaseDatos) utilizando DriverManager.getConnection(...).

8. Conclusiones

- Mediante el desarrollo del proyecto se logró recrear un sistema que permite realizar las funcionalidades principales de un CRUD.
- Se pusieron en práctica los conceptos y conocimientos adquiridos en la materia de Programación Orientada a Objetos, además se estructuraron los formularios presentes en el sistema con la ayuda de Java Swing, finalmente se logró la conexión del sistema con la base de datos MySQL.
- Esta actividad ayudó a fortalecer los conocimientos y aprendizaje en cuanto a las bases de POO como a todo lo que tiene que ver con creación de formularios, base de datos y gestionar un proyecto de software como el establecimiento de requerimientos iniciales lo que facilita la estructuración del proyecto y lo que nos ayudará a la hora de participar en proyectos más grandes a futuro.

9. Recomendaciones

- Es recomendable mantener una documentación técnica actualizada y utilizar herramientas como

GitHub para versionar el código, registrar avances y colaborar en equipo.

- Se recomienda estructurar mejor el sistema usando clases específicas para cada entidad (como Operador o Administrador) y delegar responsabilidades mediante patrones de diseño. Esto permitirá que el sistema sea más fácil de mantener, escalar y reutilizar.
- Al generar la aplicación .exe se debe verificar seleccionar versión adecuada del **JDK o JRE** correspondiente. Además, para conectar con la base de datos el archivo conector jdbc debe estar correctamente ubicado dentro del paquete en el que se va a generar el .jar o el .exe para evitar errores al ejecutarlos.