UNIVERSIDAD PERUANA LOS ANDES

FACULTAD DE INGENIERIA

ESCUELA PROFECIONAL DE INGENERIA DE SISTEMAS Y COMPUTACIÓN

TEMA

Diseño e Implementación de un sistema de login con Base de Datos para una Plataforma de Biblioteca Virual

---

CURSO: Desarrollo de Aplicaciones

DOCENTE: Ing. Fernandez Bejarano Raul Enrique

INTEGRANTES:

Ramón Puente Eduardo Fredy
Untiveros Moreno Diego Alonso

HUANCAYO - PERÚ

CICLO: IV

SECCION: A1

Contenido

1. Introducción a la Integración de Base de Datos 3
2. Arquitectura del Sistema 3
   - **2.1 Tecnologías Utilizadas 3**
3. Estructura de la Base de Datos 4
   - **3.1 Esquema de Base de Datos 4**
   - **3.2 Características Técnicas 6**
   - **3.3 Script de Creación 6**
4. Componentes del Sistema 6
   - **4.1 Estructura del Proyecto 6**
   - **4.2 Clase de Conexión: MiConexion.java 6**
5. Funcionalidades Implementadas 7
   - **5.1 Registro de Usuarios 7**
   - **5.2 Inicio de Sesión 7**
   - **5.3 Recuperación de Contraseña 8**
6. Datos de Prueba 8
7. Seguridad del Sistema 8
   - **7.1 Validación de Correo Institucional 9**
   - **7.2 Medidas de Seguridad Implementadas 9**
8. Guía de Instalación 9

8.1 Requisitos Previos **9**
8.2 Pasos de Instalación **9**
9. Evidencias de Funcionamiento 10
9.1 Pantallas del Sistema **10**
9.2 Mensajes de Sistema Verificados **11**
10. Resultados Obtenidos 12
10.1 Logros del Proyecto **13**
10.2 Beneficios Alcanzados **13**
11. Próximos Pasos y Mejoras Futuras 13
11.1 Mejoras de Seguridad **13**
11.2 Funcionalidades Adicionales **14**
12. Conclusiones 14

1. Introducción a la Integración de Base de Datos

LibConnect es una plataforma de biblioteca virtual innovadora diseñada para revolucionar el acceso al conocimiento en instituciones educativas. El presente documento detalla la implementación del sistema de autenticación y gestión de usuarios, que constituye el núcleo fundamental de seguridad y control de acceso de la plataforma.

El sistema integra una base de datos MySQL robusta que permite almacenar de forma segura la información de usuarios (estudiantes, docentes, bibliotecarios y administradores), habilitando funcionalidades como registro, autenticación, gestión de perfiles y control de acceso a recursos bibliográficos digitales.

2. Arquitectura del Sistema

2.1 Tecnologías Utilizadas

Frontend:

- Java Swing para la interfaz gráfica de usuario

- Componentes personalizados: MyTextField, MyPasswordField, Button
- Diseño responsivo con MigLayout

Backend:

- Java SE (JDK 22)
- Patrón de diseño MVC (Modelo-Vista-Controlador)
- Gestión de conexiones con JDBC

Base de Datos:

- MySQL Server 8.0
- Gestor: Navicat Premium
- Conector: mysql-connector-j-8.0.33.jar

Librerías Adicionales:

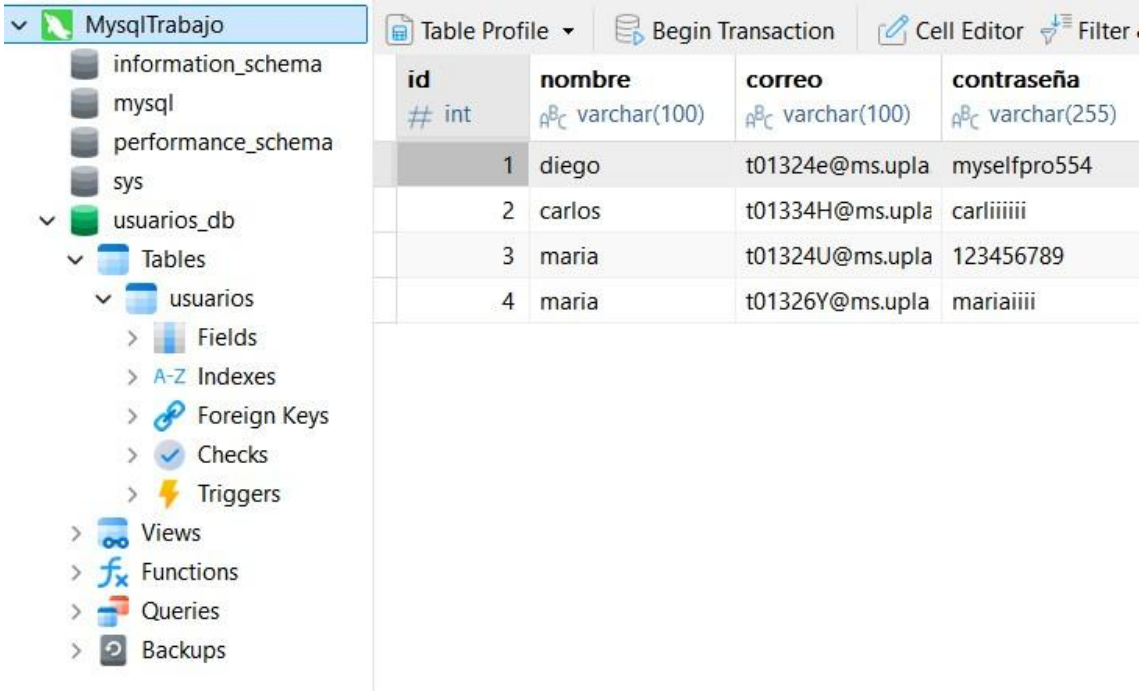
- TimingFramework-0.55.jar (para animaciones de interfaz)
- miglayout-4.0.jar (gestor de diseño avanzado)

3. Estructura de la Base de Datos

3.1 Esquema de Base de Datos

Nombre de la base de datos: usuarios_db

Tabla principal: usuarios



Campo	Tipo	Restricciones	Descripción
id	int	PRIMARY KEY, AUTO_INCREMENT	Identificador único del usuario
nombre	varchar(100)	NOT NULL	Nombre completo del usuario
correo	varchar(100)	NOT NULL, UNIQUE	Email institucional único
contraseña	varchar(255)	NOT NULL	Contraseña del usuario

3.2 Características Técnicas

- **Character Set:** utf8mb4 (soporte completo Unicode)
- **Collation:** utf8mb4_0900_ai_ci
- **Engine:** InnoDB (transacciones ACID)
- **Data Length:** 16 KB

3.3 Script de Creación

```
CREATE DATABASE usuarios_db CHARACTER SET utf8mb4
COLLATE utf8mb4_0900_ai_ci; USE usuarios_db;
CREATE TABLE usuarios ( id INT PRIMARY KEY
AUTO_INCREMENT, nombre VARCHAR(100)
NOT NULL, correo VARCHAR(100) NOT NULL UNIQUE,
contraseña VARCHAR(255) NOT NULL ) ENGINE=InnoDB;
```

4. Componentes del Sistema

4.1 Estructura del Proyecto

Organización de Paquetes:

- **com.raven.component:** Paneles principales (PanelCover, PanelLoginAndRegister)
- **com.raven.database:** Gestión de conexión (MiConexion.java)
- **com.raven.icon:** Recursos gráficos (mail.png, pass.png, user.png)
- **com.raven.main:** Clase principal (Main.java)
- **com.raven.swing:** Componentes personalizados UI

4.2 Clase de Conexión: MiConexion.java

Esta clase implementa el patrón Singleton para gestionar la conexión con MySQL de manera eficiente y segura.

Características principales:

- Instancia única de conexión (Singleton pattern)
- Gestión automática de recursos con try-catch
- Manejo robusto de excepciones SQLException

- Configuración centralizada de parámetros **Parámetros de conexión:**

```
private static final String URL =
"jdbc:mysql://localhost:3306/usuarios_db";
private static final String USUARIO = "root";
private static final String CONTRASENA =
"123456";
```

5. Funcionalidades Implementadas

5.1 Registro de Usuarios

El sistema permite registrar nuevos usuarios con validaciones exhaustivas:

Validaciones implementadas:

- ✓ Verificación de correo institucional (@ms.upla.edu.pe)
- ✓ Validación de campos no vacíos
- ✓ Formato válido de dirección email
- ✓ Prevención de registros duplicados (UNIQUE constraint)
- ✓ Longitud adecuada de contraseña

Mensajes del sistema:

- "Solo se permiten correos institucionales (@ms.upla.edu.pe)"
- "Usuario registrado correctamente"

5.2 Inicio de Sesión

Proceso de autenticación:

1. Captura de credenciales del usuario
2. Validación de formato de correo institucional
3. Consulta SQL a la base de datos
4. Verificación de contraseña
5. Generación de sesión de usuario

Mensajes de confirmación:

- "Inicio de sesión exitoso. ¡Bienvenido!"
- "Debe usar su correo institucional (@ms.upla.edu.pe)"

5.3 Recuperación de Contraseña

Funcionalidad preparada para implementación futura que incluirá:

- Validación de correo electrónico registrado
- Generación de código de verificación temporal
- Envío de email con instrucciones
- Proceso seguro de restablecimiento

6. Datos de Prueba

La base de datos incluye registros de prueba para validar el correcto funcionamiento del sistema:

ID	Nombre	Correo	Contraseña
1	diego	t01324e@ms.upla	myselfpro554
2	carlos	t01334H@ms.upla	carliiiiiii
3	maria	t01324U@ms.upla	123456789

7. Seguridad del Sistema

7.1 Validación de Correo Institucional

El sistema implementa validación estricta de correos institucionales:

```
if (!correoTexto.endsWith("@ms.upla.edu.pe")) {
JOptionPane.showMessageDialog(null, "Solo se
permiten correos institucionales."); return; }
```

7.2 Medidas de Seguridad Implementadas

- **PreparedStatements:** Prevención de inyección SQL
- **Validación de entrada:** Filtrado de datos maliciosos
- **Control de acceso:** Verificación de credenciales en BD
- **Encriptación:** Preparado para implementar hashing de contraseñas

8. Guía de Instalación

8.1 Requisitos Previos

- JDK 22 o superior instalado
- MySQL Server 8.0 o superior
- IDE Java: NetBeans, IntelliJ IDEA o Eclipse
- Navicat Premium (opcional para gestión visual)

8.2 Pasos de Instalación

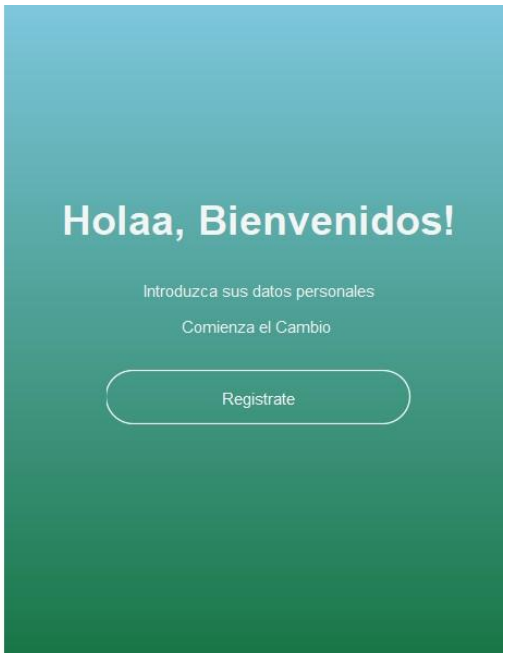
1. **Instalar MySQL Server** ○ Configurar usuario root con contraseña ○ Iniciar el servicio MySQL
2. **Crear la base de datos** ○ Ejecutar el script de creación proporcionado ○ Verificar la creación de la tabla usuarios
3. **Configurar el proyecto Java** ○ Importar las librerías JAR al proyecto ○ Verificar la ruta de

conexión en MiConexion.java ○ Ajustar credenciales si es necesario

4. **Compilar y ejecutar** ○ Build del proyecto en el IDE ○ Ejecutar la clase Main.java

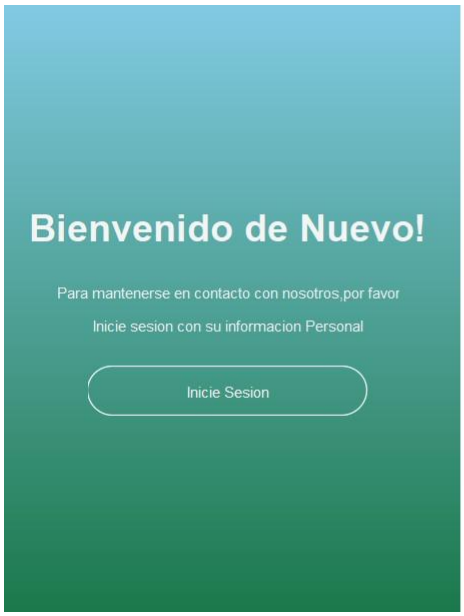
9. Evidencias de Funcionamiento

9.1 Pantallas del Sistema



Interfaz de Inicio de Sesión:

- Campo de correo electrónico con ícono
- Campo de contraseña con seguridad
- Botón "Iniciar Sesión" con diseño moderno
- Enlace de recuperación de contraseña
- Panel lateral de bienvenida con degradado



Interfaz de Registro:

- Campo de nombre de usuario
- Campo de correo institucional
- Campo de contraseña segura
- Botón "Registrar" con animación
- Validación en tiempo real

9.2 Mensajes de Sistema Verificados

Registro exitoso:

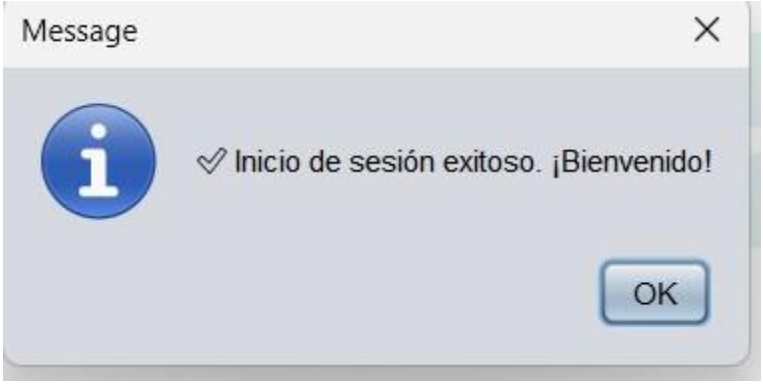
"Usuario registrado correctamente"

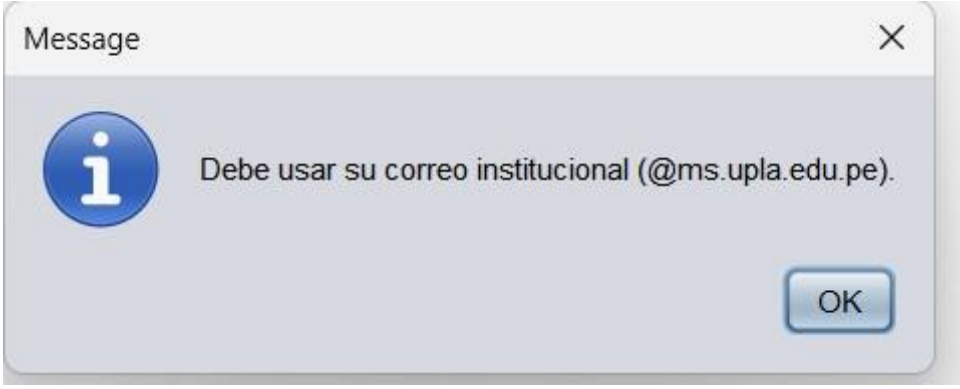
Login exitoso:

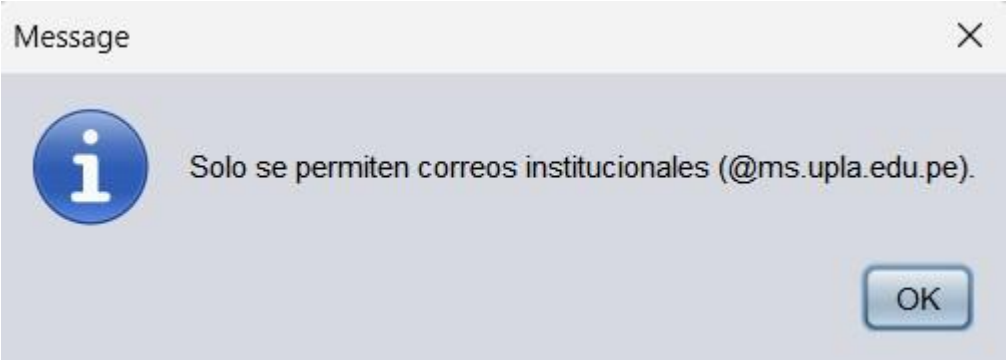
"Inicio de sesión exitoso. ¡Bienvenido!"

Validaciones:

- "Solo se permiten correos institucionales (@ms.upla.edu.pe)"
- "Debe usar su correo institucional (@ms.upla.edu.pe)"







10. Resultados Obtenidos

10.1 Logros del Proyecto

- ✓ **Persistencia de datos:** Almacenamiento permanente de usuarios
- ✓ **Autenticación funcional:** Sistema de login operativo
- ✓ **Validaciones robustas:** Control de correos institucionales
- ✓ **Interfaz profesional:** Diseño moderno y amigable
- ✓ **Arquitectura escalable:** Preparado para nuevos módulos
- ✓ **Integración completa:** Frontend-Backend-Database

10.2 Beneficios Alcanzados

Para User:

- Acceso seguro con credenciales institucionales o Con dominio
- Registro simplificado y rápido
- Interfaz intuitiva y moderna **Para user:**
- Control de usuarios registrados
- Base para gestión de libros
- Sistema confiable y estable

Para la Admin:

- Digitalización del proceso de autenticación
- Seguridad en el acceso a la plataforma
- Reducción de errores en gestión de usuarios
- Trazabilidad y auditoría de accesos

11. Próximos Pasos y Mejoras Futuras

El sistema está preparado para incorporar las siguientes mejoras:

11.1 Mejoras de Seguridad

- Implementación de encriptación BCrypt para contraseñas
- Sistema de tokens JWT para sesiones

- Autenticación de dos factores (2FA)
- Logs de auditoría de accesos

11.2 Funcionalidades Adicionales

- Recuperación de contraseña por email funcional
- Sistema de roles (User,admin)
- Panel de administración de usuarios
- Perfil de usuario editable

12. Conclusiones

Este proyecto constituye la base fundamental para una plataforma bibliotecaria digital integral, lista para ser expandida con los módulos adicionales planificados. El sistema mantiene altos estándares de calidad, seguridad y usabilidad, estableciendo un precedente para futuros desarrollos de sistemas de información en la institución.

La experiencia adquirida en el desarrollo de LibConnect proporciona habilidades valiosas para la creación de soluciones tecnológicas que respondan a necesidades reales de las bibliotecas modernas, preparando a los desarrolladores para enfrentar desafíos similares en su carrera profesional.

Competencias desarrolladas:

- Diseño e implementación de bases de datos relacionales
- Programación orientada a objetos en Java
- Desarrollo de interfaces gráficas con Swing
- Integración Frontend-Backend-Database
- Implementación de validaciones y seguridad
- Trabajo en equipo y documentación técnica

Fecha de elaboración: Octubre 2025