



Submitted BY:-

- Eaman Fatima
- Nida Akram
- Syeda Abeeha Batool
- Khadijaa Aftaab

Roll Numbers:-

- 2k23-BSCS-443
- 2k23-BSCS-409
- 2k23-BSCS-436
- 2k23-BSCS-414

Section:-

Yellow(Y)

Project Title:-

Developing a library management system(LMS).

Library Management System(LMS)

Documentation

Project Overview

This Library Management System (LMS) is a desktop application developed using Java Swing for the graphical user interface (GUI) and MySQL for database management. It facilitates efficient handling of library operations, including book management, user authentication, staff management, and borrowing transactions.

Tools & Technologies

<u>Tools</u>	<u>Purposes</u>
Java	Core programming language
Swing	GUI framework for building desktop applications
MySQL	Relational database management system
JDBC	Java Database Connectivity for database interaction

Application Structure

1. Main.java

- **Function:**

Initializes the application and displays the login form.

2. DBConnect.java

- **Function:**

Establishes a connection between Java and the MySQL database using JDBC.

3. LoginForm.java

- **Function:**

Provides a login interface with email and password fields.

- **Features:**

- Toggle password visibility
- Validate credentials against the database

4. SignupForm.java

- **Function:**

Allows new users to register by providing their name, email, and password.

- **Features:**

- Saves user information to the `users` table

5. Books Management

- **Function:**

Central controller for book-related operations.

- **AddBook.java:** Interface to add new books to the library.
- **ViewBooks.java:** Displays a list of all books in a table format.
- **UpdateBook.java:** Allows editing of existing book details.
- **DeleteBook.java:** Facilitates removal of books by ID.

6. Staff Management

- **Function:**

Manages staff data, including adding and updating staff information.

7. Borrowing System

- **Function:**

Handles book borrowing transactions by students.

- **Features:**

- Records borrow date, due date, return date, and fine amount
- Tracks borrowing status (e.g., pending, overdue)

Database Structure

Database name: **library_db**

TABLES DETAILS:

1.Users

Column	Data Type	Description
user_id	INT	Primary Key
name	VARCHAR	User's full name

Column	Data Type	Description
--------	-----------	-------------

email	VARCHAR	User's email (unique)
password	VARCHAR	Hashed password

2. students

Column	Data Type	Description
student_id	INT	Primary Key
name	VARCHAR	Student's full name
email	VARCHAR	Student's email
phone	VARCHAR	Contact number
course	VARCHAR	Enrolled course
department	VARCHAR	Department name
registration_date	DATE	Date of registration

3. books

Column	Data Type	Description
book_id	INT	Primary Key
title	VARCHAR	Book title
author	VARCHAR	Author's name
publisher	VARCHAR	Publisher's name
isbn	VARCHAR	ISBN number
genre	VARCHAR	Book genre
quantity	INT	Total copies available
available	INT	Copies available for borrowing
published_year	INT	Year of publication

4. borrowings

Column	Data Type	Description
borrow_id	INT	Primary Key
student_id	INT	Foreign Key (students)
book_id	INT	Foreign Key (books)
vvborrow_date	DATE	Date when book was borrowed
due_date	DATE	Due date for return
return_date	DATE	Actual return date
fine	DECIMAL	Fine amount incurred
status	VARCHAR	Current status (e.g., returned, borrowed)

5. staff

Column	Data Type	Description
staff_id	INT	Primary Key
name	VARCHAR	Staff member's full name
email	VARCHAR	Staff member's email
phone	VARCHAR	Contact number
position	VARCHAR	Job position (e.g., librarian)
department	VARCHAR	Assigned department
date_joined	DATE	Date of joining

Relationships

- `borrowings.student_id → students.student_id`
 - `borrowings.book_id → books.book_id`
-

Key Features

- **User Authentication:** Secure login and signup system with password encryption.
 - **Book Management:** Add, view, update, and delete book records.
 - **Staff Management:** Manage staff information and roles.
 - **Borrowing System:** Track book borrowings, due dates, returns, and fines.
 - **Database Integration:** All data stored in a MySQL database for persistence.
-

✓ ER-Diagram



