PROJECT: BOOK MANAGEMENT SYSTEM

Table of Contents

Aim/Overview of the project:

Book Management System – A web-based application that allows users to register, login, and manage their personal book collection with CRUD functionality.

2. Objective

The main goal is to create a simple and secure platform where users can manage their own books. Each user has their own book records, with no interference from other users.

3. Technology Stack

- Frontend: HTML, CSS, JSP, Bootstrap
- Backend: Java Servlets
- Database: MySQL
- Server: Apache Tomcat
- IDE: Eclipse / IntelliJ

4. Features of the Application

- User Registration and Login
- Add New Books
- View Books
- Edit or Delete Own Books
- Secure User-based Access Control
- Clean UI with Success Alerts

5. System Requirements

- OS: Windows/Linux
- Software: JDK 8+, Apache Tomcat, MySQL
- Browser: Chrome, Firefox
- RAM: Minimum 2GB

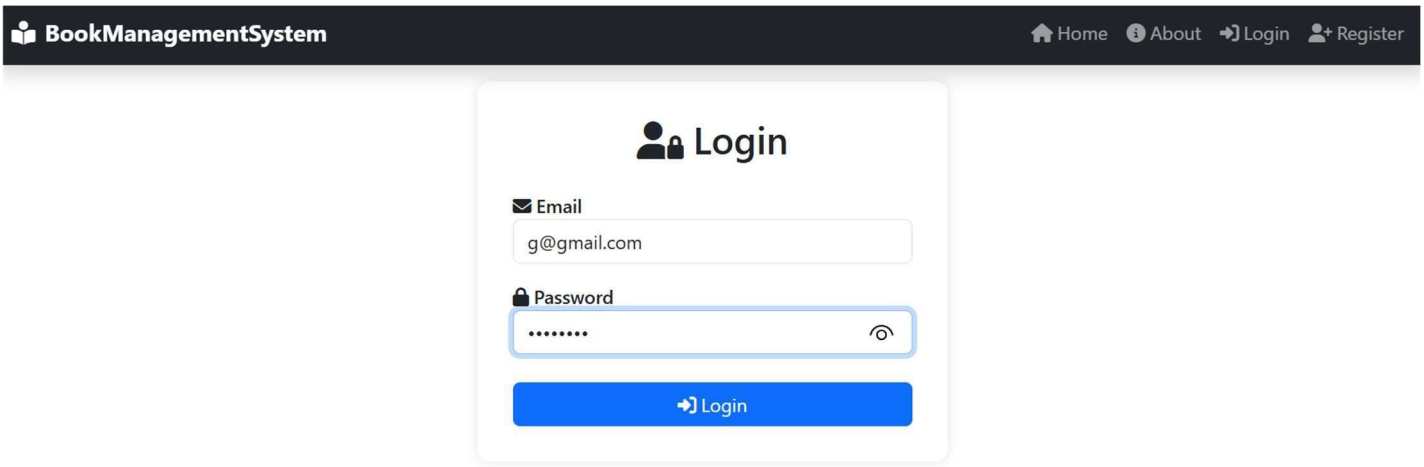
6. Database Design

- Users Table: Stores user details like id, username, email, password
- Books Table: Stores book details with foreign key user_id to associate each book with the correct user.

7. Module Description

7.1 User Authentication

Handles registration and login using email and password. After login, session is created.



7.2 Add Book

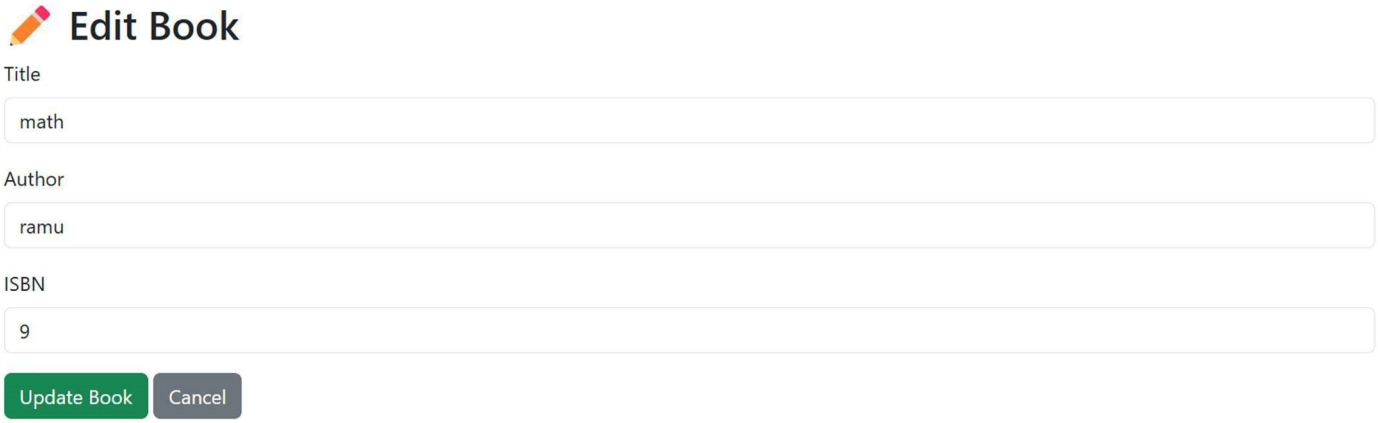
Logged-in users can add new books by filling title, author, and ISBN.

7.3 View Books

Shows only those books which are added by the logged-in user.

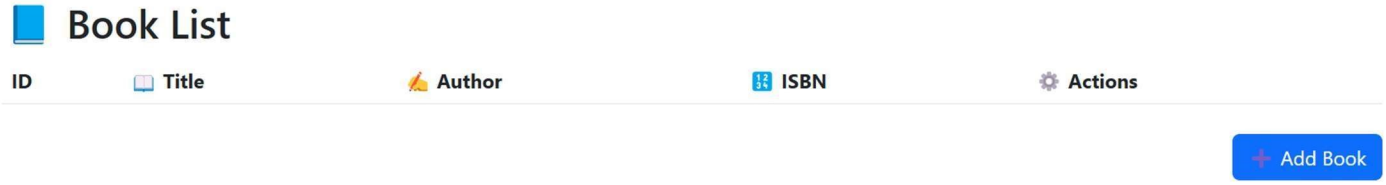
7.4 Update Book

User can update only their own book entries.



7.5 Delete Book

Logged-in users can delete only their own books from the database.



8. Access Control

The application ensures that only the authenticated user can access, edit, or delete their own books using session-based control.

Below is the MySQL database “book_management” consisting of table “books” for adding, updating and deleting information:

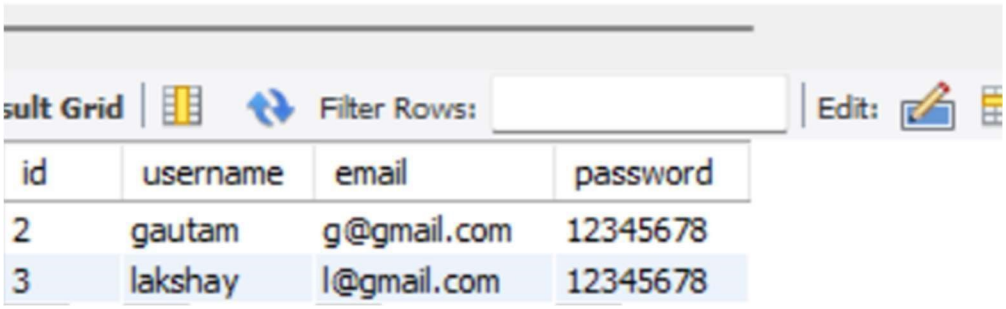
```sql
CREATE DATABASE book_management;
USE book_management;
CREATE TABLE books (
    id INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(100) NOT NULL,
    author VARCHAR(100) NOT NULL,
    isbn VARCHAR(20) NOT NULL UNIQUE
);
select * from books;
ALTER TABLE books ADD email VARCHAR(255);
```

Result Grid | Filter Rows: | Edit:

id	title	author	isbn	email
10	math	ramu	9	g@gmail.com

Another table “users” is used to store the credentials of the signup user and check the validation when user try to login to access their account:

```
4 • CREATE TABLE users (
5       id INT AUTO_INCREMENT PRIMARY KEY,
6       username VARCHAR(50) NOT NULL UNIQUE,
7       password VARCHAR(100) NOT NULL
8   );
9 • select * from users;
```



sult Grid | Filter Rows: | Edit:

id	username	email	password
2	gautam	g@gmail.com	12345678
3	lakshay	l@gmail.com	12345678

9. User Interface Overview

- Clean Bootstrap-based UI
- Navigation bar changes based on login/logout
- Alert messages for operations like Add, Update, Delete
- Simple forms for book management

10. Working Flow of the System

- User registers
- Logs in → session created

- Access to Add/View/Update/Delete Books
- Only logged-in user's data is visible
- Logout destroys session and restricts access

11. Conclusion

The Book Management System project successfully demonstrates the use of Java, JSP, Servlets, and MySQL to create a simple yet secure web application with real-world utility. It allows users to register, log in, and manage their personal collection of books efficiently. By implementing session management, it ensures that each user can only view, update, or delete their own books, which adds a strong layer of personalization and security.

This project not only strengthened understanding of backend connectivity using JDBC, but also improved practical skills in creating a modular, scalable, and user-friendly application using core Java technologies.

12. Future Scope

- Admin Panel for managing users
- Book categories and search functionality
- Password reset system
- RESTful APIs for mobile app integration

13. References

- Java Servlet Documentation
- MySQL Documentation
- Bootstrap 5 Docs
- Stack Overflow
- GeeksforGeeks

Github link: https://github.com/Lakshya3000/AIP_Project