

Library Management System

A PROJECT REPORT

Submitted by

Nishant Rejra (23BCS12961)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE & ENGINEERING



Chandigarh University

Nov, 2025



BONAFIDE CERTIFICATE

Certified that this project report "**Library Management System**" is the bonafide work of "**Ayush Kohli**" who carried out the project work under my supervision.

SIGNATURE

Dr. Sandeep Singh Kang

SIGNATURE

BATCH HEAD

BE-CSE

SUPERVISOR

BE-CSE

TABLE OF CONTENTS

| S. No. | Particulars | Page No. |
|---------------|-------------------------------|-----------------|
| 1. | Project Name | 1 |
| 2. | Reference Website Link | 2 |
| 3. | Project Description | 3 |
| 4. | Problem Statement | 4 |
| 5. | High Level Design | 5 |
| 6. | Key Features | 6 |
| 7. | System Flow (Summary) | 7 |
| 8. | Flow Chart | 8 |
| 9. | Future Scope | 9 |
| 10. | Conclusion | 10 |



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Name: Nishant Rejra
Branch: BE-CSE
Subject name: Full Stack - 1

UID: 23BCS12961
Section/Group: KRG-3B
Subject code: 23CSH-339

1. PROJECT NAME

Library Management System

2. REFERENCE WEBSITE LINK

<https://spring.io/projects/spring-boot>

Other References:

- React Documentation (react.dev)
- MySQL Official Docs (<https://dev.mysql.com/doc>)
- Axios (<https://axios-http.com>) for frontend-backend integration

3. PROJECT DESCRIPTION

The Library Management System is a full-stack web application built using React.js (frontend) and Spring Boot (backend) with a MySQL database. It is designed to automate and digitalize all library operations — such as managing books, members, and book issues — through a single interactive platform.

The system enables an Admin to securely log in, add/update/delete books and members, and record when books are issued or returned. The frontend provides a clean and responsive interface, while the backend ensures secure and efficient data handling through RESTful APIs.

Tech Stack Overview:

| Layer | Technology | Description |
|-----------|-------------------------------|--|
| Frontend | React.js, Axios | User interface, API requests, and navigation |
| Backend | Spring Boot (Java), REST APIs | Handles business logic and database operations |
| Database | MySQL | Stores all book, member, and issue records |
| API Layer | REST endpoints | Connects frontend and backend |

4. PROBLEM STATEMENT

Traditional library systems rely on manual record-keeping, leading to:

Users need a unified dashboard that:



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

- Time-consuming processes
- Data inaccuracy and redundancy
- Difficulty tracking issued or returned books.

The system ensures **fast, reliable, and easy-to-use** operations through a modern full-stack web application architecture.

5. HIGH LEVEL DESIGN

Frontend

- **Technology:** React.js, HTML5, CSS3, JavaScript (ES6+), Axios
- **Features:**
 - Admin Login Page
 - Book Management Module (Add, Edit, Delete, View)
 - Member Management Module (Add, Edit, Delete, View)
 - Issued Books Module (Issue / Return functionality)
 - Responsive and user-friendly layout

Backend

- **Technology:** Java Spring Boot (v3.x)
- **Endpoints:**
 - /admin/login → Validate admin credentials
 - /books → Get/Add/Edit/Delete books
 - /members → Manage member data
 - /issued-books → Manage issue/return transactions
- **Responsibilities:**
 - Handle CRUD operations through REST APIs
 - Validate admin login
 - Communicate with MySQL database
 - Return JSON responses to React frontend

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Database (MySQL)

Tables:

1. admin – stores admin login credentials
2. books – stores book information
3. members – stores library members' details
4. issued_books – records issued and returned books

6. Key Features

User Features

- **Admin Login:** Secure login authentication for admin access
- **Book Management:** Add, view, update, and delete book details
- **Member Management:** Add and maintain member records
- **Issue/Return Management:** Track issued and returned books with live updates
- **Automatic Quantity Update:** Reduces or increases stock when books are issued/returned

Technical Features

- REST API-based communication between frontend and backend
- Axios integration for sending HTTP requests from React
- Entity-Repository-Service-Controller architecture in Spring Boot
- Automatic database table creation using JPA
- Modular, maintainable full-stack design
- Cross-Origin Resource Sharing (CORS) for React–Spring Boot connection

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

7. SYSTEM FLOW

Step-by-Step Flow:

1. Admin Login:

- Admin enters credentials in Login.js
- React sends POST request to /admin/login
- Spring Boot verifies credentials using AdminService and database

2. Book Management:

- Admin can view all books (/books)
- Add a new book using form → POST request → backend saves in MySQL
- Edit or delete books directly from UI

3. Member Management:

- Admin can view, add, or update member details
- All data saved in members table\

4. Issue Book:

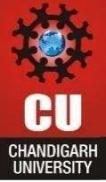
- Admin selects member and book → sends POST to /issued-books/{bookId}/{memberId}
- Backend reduces book quantity and adds issue record in issued_books

5. Return Book:

- PUT request to /issued-books/{issueId}/return
- Backend updates record, marks “returned = true” and restores quantity

6. Frontend Updates:

- React dynamically updates all lists (Books, Members, Issued Books)
- No manual page reload required



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

8. FUTURE SCOPE

1. JWT Authentication:

Implement secure token-based login for Admins and Members.

2. Role-Based Access:

Add different user roles (Admin, Member, Librarian).

3. Fine Calculation:

Automatically calculate fines for late book returns.

4. Email Notifications:

Send alerts/reminders for due or overdue books.

5. Search and Filter:

Implement live search and category-based filtering.

6. Dashboard Analytics:

Show total books, issued books, and members using charts.

7. Cloud Deployment:

Host the application on AWS/Render for live use.

8. Dark Mode UI:

Add theme toggle for better UX.

9. Conclusion

The Library Management System successfully demonstrates a full-stack implementation of a modern CRUD-based web application using React, Spring Boot, and MySQL.

It automates library operations and provides a secure, responsive, and user-friendly admin interface.

This project highlights practical skills in:

- REST API development
- Database integration with JPA
- React-based dynamic UI
- Full-stack data flow and architecture

The system is scalable, extendable, and provides a solid foundation for future enhancements like role-based security and analytics dashboards.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Screenshots:

Admin Login

Username

Password

Login

Library Management [Books](#) [Add Book](#) [Members](#) [Add Member](#) [Issued Books](#) [Issue Book](#)

Books List

[Add Book](#)

| ID | Title | Author | Publisher | Category | ISBN | Published Date | Total Copies | Available Copies | Actions |
|----|-------------------|-----------------------|--------------------|---------------------|---------------|----------------|--------------|------------------|---|
| 6 | Godaan | Munshi Premchand | Rajpal and Sons | Novel | 9788126715025 | 1936-01-01 | 10 | 8 | Edit Delete |
| 7 | Ghalib Ki Shayari | Mirza Ghalib | Indian Press | Poetry | 9788171672678 | 1869-06-01 | 6 | 5 | Edit Delete |
| 8 | Raag Darbari | Shrilal Shukla | Rajkamal Prakashan | Satire | 9788126712147 | 1968-09-01 | 12 | 10 | Edit Delete |
| 9 | Aag Ka Darya | Qurratulain Hyder | Rajkamal Prakashan | Historical Novel | 9788126715247 | 1959-04-15 | 8 | 6 | Edit Delete |
| 10 | Chitralekha | Bhagwati Charan Verma | Rajpal and Sons | Philosophical Novel | 9788170287934 | 1934-11-10 | 9 | 7 | Edit Delete |
| 11 | SUNRISE | Nishant | Nishant | Poet | demo | 2004-12-27 | 10 | 8 | Edit Delete |
| 12 | Shivam | s | sdemo | demo | www | 4201-02-25 | 10 | 8 | Edit Delete |

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Library Management Books Add Book Members Add Member Issued Books Issue Book

Add New Book

Title

Author

Publisher

Category

Isbn

Published Date

 dd-mm-yyyy

Total Copies

Available Copies

Library Management Books Add Book Members Add Member Issued Books Issue Book

Members List

| ID | Name | Email | Phone | Address | Actions |
|----|---------------|-----------------------------|------------|-----------------|---|
| 15 | Aarav Sharma | aravttt.sharma2@example.com | 9876543220 | Delhi | <input type="button" value="Edit"/> <input type="button" value="Delete"/> |
| 16 | Zara Khan | zara.khan2@example.com | 9876543221 | Lucknow | <input type="button" value="Edit"/> <input type="button" value="Delete"/> |
| 17 | Rahul Verma | rahul.verma2@example.com | 9876543222 | Jaipur | <input type="button" value="Edit"/> <input type="button" value="Delete"/> |
| 18 | Faiz Ahmad | faiz.ahmad2@example.com | 9876543223 | Bhopal | <input type="button" value="Edit"/> <input type="button" value="Delete"/> |
| 19 | Ananya Singh | ananya.singh2@example.com | 9876543224 | Patna | <input type="button" value="Edit"/> <input type="button" value="Delete"/> |
| 20 | Nishant Rejra | nishantrejra@gmail.com | 7894564567 | demo | <input type="button" value="Edit"/> <input type="button" value="Delete"/> |
| 21 | John Doe | john@example.com | 9876543210 | 123 Main Street | <input type="button" value="Edit"/> <input type="button" value="Delete"/> |

Issued Books

| ID | Book Title | Member Name | Issue Date | Return Date | Actions |
|----|--------------|--------------|------------|-------------|---------------------------------------|
| 1 | Raag Darbari | Aarav Sharma | 2025-11-09 | 2025-11-09 | <input type="button" value="Delete"/> |
| 2 | Aag Ka Darya | Rahul Verma | 2025-11-10 | 2025-11-10 | <input type="button" value="Delete"/> |