Business Requirements Document

Table of Contents

[1. Business Objectives 2](#_cb6pswn95csr)

[2. Target Users 2](#_ppqohfd8jvpw)

[Librarians 2](#_225q2xaldlpl)

[Students and Faculty 2](#_f2k5ndb3o5u8)

[3. Functional Requirements 2](#_9l5ch8ippxi)

[User Management 2](#_tyens65ffowp)

[Book Management 2](#_jrlzptpx70fk)

[Borrowing and Returning 3](#_3u4oz7svlrzi)

[Reporting and Analytics 3](#_gluylc5qb720)

[4. Non-Functional Requirements 3](#_8cr82vqy8l3m)

[Performance 3](#_pibmfp1tcj3)

[Security 3](#_x68nthf0seb3)

[Usability 3](#_kt339mahdf2g)

[5. Project Deliverables 3](#_e8gqmpmkgv4m)

[Backend 3](#_ehiwgehrou04)

[Frontend 4](#_wfjfawvorbyx)

[6. Project Success Criteria 4](#_c3i24uizdjvm)

# 

# **Library Management System**

The Library Management System (LMS) project aims to develop a robust and comprehensive system to efficiently manage library resources. Utilizing Spring Boot for the backend, ReactJS for the frontend, and MySQL as the database, the system will facilitate streamlined book cataloging, user management, and borrowing activities.

### **1. Business Objectives**

The primary objective of the LMS is to enhance the management of library resources for various stakeholders, including librarians, students, and faculty. Key objectives include:

* Efficient management of books, user accounts, and borrowing processes.
* Improved user experience for librarians, students, and faculty members.
* Streamlined book categorization, borrowing, and return processes.

### **2. Target Users**

#### Librarians

Manage library collections, user accounts, and oversee borrowing and return processes.

#### Students and Faculty

Access the library catalog, search for books, and place borrowing requests.

### **3. Functional Requirements**

#### User Management

Create, edit, and delete user accounts.

Manage user roles and permissions.

Track user borrowing history.

#### Book Management

Add, edit, and delete book records.

Categorize books using genres and subjects.

Maintain book availability status.

#### Borrowing and Returning

Allow users to search for available books.

Facilitate borrowing requests for available books.

Manage borrowing periods and due dates.

Enable book returns and update availability.

#### Reporting and Analytics

Generate reports on book usage and borrowing trends.

Analyze user borrowing patterns.

Track overdue books and generate notifications.

### **4. Non-Functional Requirements**

#### Performance

Support multiple concurrent users without performance degradation.

Ensure efficient and responsive book search and filtering operations.

#### Security

Implement secure user authentication and authorization mechanisms.

Protect sensitive user information and borrowing records.

Prevent unauthorized access to library resources.

#### Usability

Provide a user-friendly interface that is easy to navigate and understand.

Offer clear instructions and guidance for users.

Implement consistent design patterns across frontend and backend.

### **5. Project Deliverables**

#### Backend

* Spring Boot application with RESTful APIs.
* Database schema and data migration scripts.
* Unit tests for backend components.

#### Frontend

* ReactJS application with user interface components.
* Routing and navigation between different pages.
* Integration with backend APIs for data fetching and manipulation.

### **6. Project Success Criteria**

* Fulfillment of all functional and non-functional requirements.
* Successful deployment and adoption by librarians, students, and faculty members.
* Positive user feedback regarding ease of use and functionality.
* Demonstrated improvement in library management efficiency.