# **Software Requirements Specification (SRS)**

**Library Management System (LMS)**

### **Introduction**

This document outlines the functional and non-functional requirements for the Library Management System (LMS). The LMS will help in library operations, including managing books, members, borrowing/returning, and generating reports. It will ensure a user-friendly interface for librarians and library members to interact with the system effectively.

### **Definitions, Acronyms, and Abbreviations**

* **LMS:** Library Management System
* **User:** A library member who can borrow books or other resources.
* **Librarian:** A staff member responsible for managing the library’s resources and user accounts.
* **UI:** User Interface

### **System Requirements**

#### ***Functional Requirements***

#### **User Authentication and Authorization**

**Description**: The system must allow users to register, log in, and manage their accounts.

FR1: User can register with a username, email, and password.

FR2: Registered user can log in to the system

FR3: Registered user can manage their own details

#### **Book Search and Discovery**

**Description:** Users and librarians must be able to search for books by various filters (e.g., title, author, ISBN, genre).

FR4: Search by title, author, ISBN, genre, or availability.

#### **Borrowing and Returning Books**

**Description:** Users should be able to borrow and return books.

FR5: Users can borrow book.

FR6: Users can return book.

#### **Book Management**

**Description**: Librarians will be able to add, update, delete, and view books in the system.  
FR7: Librarians can add new books.

FR8: Librarians can update book information.

FR9: Librarians can u delete books from the system.

#### **Overdue Book Notifications**

**Description:** The system will notify users about overdue books when user login.

FR10: Users will receive notification for overdue books when user login.

#### ***Non-Functional Requirements***

* The UI must be user-friendly.
* The system must be able to support at least 100 concurrent users without noticeable performance degradation.
* The system should respond to user queries and actions within 3 seconds.

### **System Constraints**

* **Constraints:**
* The backend is built on Spring Boot with Maven.
* The database used is MySQL 8+
* The frontend is built with React 18