# Project Specification: Library Book Management System

### Brief Description:

* The library book management system allows users to search, borrow, and return books.
* Administrators can add new books, edit book information, and track borrowings.
* The system supports book categorization, user reviews, and author management.

### Functionalities:

1. User Functionalities:

* User Registration and Login: Users can register and log into the system.
* View Available Books: Users can search for books by title, author, category, and publication year.
* View Book Details: Display detailed information about a selected book, including user reviews.
* Book Reservation: Users can borrow or reserve books.
* View Borrowings: Users can see a list of books they have borrowed and the due dates for returns.
* Return Book: Users can mark books as returned.
* Rate and Review Books: Users can rate and write reviews for books they have read.

1. Administrative Functionalities:

* Add New Books: Administrators can add new books to the system.
* Edit Existing Books: Administrators can edit book information.
* Delete Books: Administrators can remove books from the system.
* View All Borrowings: Administrators can track which books are borrowed and who borrowed them.
* Manage Categories: Administrators can add, edit, and delete book categories.
* Manage Authors: Administrators can add, edit, and delete author information.

### Technical Specification:

1. Backend (Spring Boot):

* Spring Boot: Framework for creating RESTful APIs.
* Spring Security: For user authentication and authorization.
* JPA/Hibernate: For database interactions.
* MySQL: Database for storing user, book, review data, etc.
* Spring Data JPA: For easy data access.

1. Frontend (Angular):

* Angular: Framework for building SPA (Single Page Application).
* Angular Material: For UI components.
* Reactive Forms: For form management and validation.
* HTTP Client: For communication with backend API.
* Routing: For navigation within the application.
* Local Storage Mechanism: For storing JWT tokens to maintain user sessions and ensure application resilience on page refresh.

### Application Structure:

1. Backend (Spring Boot):

* Entities: User, Role, UserRole, Book, Category, Author, Borrowing, Review.
* Controllers: Endpoints for books, users, borrowings, categories, authors, and reviews.
* Services: Business logic.
* Repositories: Database access.
* DTOs: Data Transfer Objects for transferring data between frontend and backend.

1. Frontend (Angular):

* Components: To display different parts of the application (book list, book details, registration, login, reviews).
* Services: For communication with backend API.
* Modules: To organize different parts of the application.
* Route Guards: To protect routes that require authentication.