



Library Management System

Develop a comprehensive Library Management System that allows users to manage book inventories, track borrower details, and handle transactions efficiently.

Features Requirements

1. User Management

- Login/Logout functionality for Admin and Users.
- Role-based access control: Admin, Librarian, and User roles.

2. Book Inventory Management

- Add, update, delete, and search for books.
- Book details: ISBN, title, author, publisher, year, genre, quantity.
- Real-time availability status.

Note: You can use the Google Books API to fetch the data to add new books (enter the book ISBN number to fetch the book details)

<https://www.googleapis.com/books/v1/volumes?q=isbn:9781787123427>

3. Borrowing System

- Checkout process for borrowing books.
- Return process including due dates and late fees calculation.
- History tracking for each user's borrowed and returned books.

4. Search and Recommendations

- Advanced search options (by title, author, genre, etc.).
- Book recommendations based on user history or popular trends.

5. Notifications and Alerts

- Email or SMS notifications for due dates, new arrivals, etc.
- Alerts for overdue books and outstanding fees.

6. Reporting

- Generate reports on book usage, overdue items, user activity, etc.
- Dashboard for admins and librarians to see real-time statistics.

Submission Requirements

- A working prototype of the Books Library Management System.
- Access to a repository with complete source code for evaluation.
- A brief presentation or video demo outlining key features and technologies used.
- A README file with setup instructions and system requirements.



Evaluation Criteria

1. **Functionality:** How well does the application meet the needs as described in the features?
2. **User Interface/User Experience:** Cleanliness of the UI and ease of navigation.
3. **Creativity and Innovation:** Originality of the approach, including unique features and problem-solving.
4. **Code Quality:** Readability, structure, and documentation of the code.
5. **Scalability and Performance:** Efficiency of the system under load, and potential for future expansion.
6. **Security:** Implementation of secure authentication, data protection, and other security best practices.