Project Report

Modern Application Development II

Kislay Ghosh   21f1003331

Library Management System - V2

About the Project
The Library Management System - V2 is a multi-user application designed for managing e-books within a library. The system facilitates the issuance, granting/revoking, and maintenance of e-books across various sections, providing a comprehensive solution for library management.

Technologies Used:

- **API**: Flask
- **Database**: SQLite
- **Frontend**: VueJS
- **Styling**: Bootstrap
- **Caching**: Redis
- **Batch Jobs**: Redis & Celery

Roles:

Librarian:

- o Maintains sections and e-books.
- o Monitors user statistics and library activities.
- o Manages e-book access and performs CRUD operations on sections and e-books.

General User:

- o Accesses and requests e-books.
- o Can request up to 5 e-books at a time.
- o Can update their profile and rate e-books.

Core Functionalities:

Authentication:

- Login/Register form for librarians and users.
- Role-based access control using Flask security or JWT-based authentication.
- Unique librarian role with admin privileges.

Librarian Dashboard:

- Displays statistics like active users, grant requests, e-books issued, and revoked.
- Automatic creation of the librarian upon database initialization.

Section Management:

- Create, update, and delete sections.
- Approval process for section modifications.

E-book Management:

1.

- CRUD operations for e-books.
- Allocation of e-books to sections.
- Approval process for e-book management.

2.

Search Functionality:

3.

- Search for e-books and sections based on various criteria.

4.

General User Functionalities:

5.

- Request and return e-books.
- Access e-books for a specified period.

o   Feedback mechanism for e-books.

6.

Librarian Functionalities:

7.

o   View and manage issue/return requests.
o   Grant or revoke e-book access.
o   Monitor library statistics.

8.

Backend Jobs:

9.

o   **Daily Reminders**: Send reminders to users who haven't interacted with the platform recently.
o   **Monthly Activity Report**: Generate and send a report summarizing library activities.
o   **CSV Export**: Provide an option for librarians to export e-book details in CSV format.

10.

Performance and Caching:

11.

o   Implement caching to enhance API performance.
o   Set cache expiry policies to manage data freshness.

Recommended Functionalities:

- **PDF Reports**: Generate and download monthly activity reports as PDFs.
- **E-book Purchases**: Allow users to purchase e-books as PDFs.
- **Responsive UI**: Design a unified and responsive user interface.
- **Frontend Validation**: Implement validation for form fields.

Optional Functionalities:

- **Styling and Aesthetics**: Enhance the frontend using Bootstrap or custom CSS.
- **Login System**: Use Flask extensions for secure login management.
- **Payment Portal**: Implement a dummy payment portal for e-book purchases.
- **Additional Features**: Integrate any other features that improve the application.

Project Folder Structure:

- **main.py**: Python code to start the application.
- **templates**: HTML files for rendering content.
- **static**: JavaScript files for frontend functionality.
- **instance**: SQLite database.

- **application**: Python code for API endpoints.
- **requirements.txt**: List of required Python packages.
- **README.md**: Project information and setup instructions.
- **API Performance.xlsx**: Performance metrics of APIs before and after caching.

About Models:

- **User**: Stores user information.
- **Role**: Defines user roles (Admin, Store Manager, User).
- **RoleUsers**: Associates users with their roles.
- **Category**: Stores categories of e-books.
- **Product**: Contains product details.
- **Cart**: Manages users' shopping carts.
- **Store**: Information about stores.
- **Store_Application**: Users applying to start a store.
- **Orders**: Details of products ordered.
- **Request**: User requests for category modifications.

Project Video Presentation: https://drive.google.com/file/d/1938L8ZEb-ZZphMe2IztBomP-4xNa9VAk/view?usp=drive_link