

Tic Tech Toe 24

Team Name : Code Blunders

1. Introduction and Overview

Our e-library system is designed to provide students with the ability to upload, access, and manage study materials and books. The system allows users to rate books, mark them as favourites for quick access, and enables the administrator (librarian) to manage users efficiently. We have focused on building a secure, scalable, and user-friendly platform, with key features including user authentication, document access control, and CRUD operations. A recommendation system is also under development, using feedback and ratings to suggest relevant study materials.

2. Login System and Secure User Authentication

We have implemented a robust user authentication system, ensuring data security through the following mechanisms:

- Login & Registration: Users can log in with their credentials or register as new users. During registration, we store essential user information securely, utilising encryption for sensitive data.
- Forgot Password Feature: We provide a password recovery option, allowing users to reset their password securely via email verification.
- Authentication Tokens: To maintain session security, we use authentication tokens that expire after a set duration, protecting the system from unauthorised access.

The image displays three screenshots of the e-library application interface:

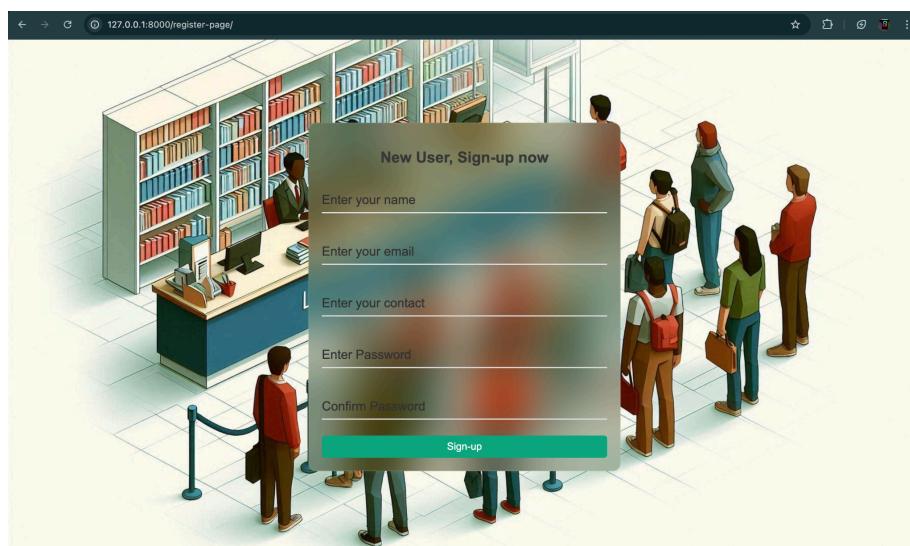
- Screenshot 1: Home Page** (Left): Shows a search bar at the top with placeholder text "Search for books, authors, or genres...". Below it is a "Popular Books" section featuring six book thumbnails with titles like "Book Title 1" and "Book Title 6". Further down is a "Recently Uploaded Books" section with six more thumbnails.
- Screenshot 2: Login Page** (Middle): A modal window titled "Sign in to our पुस्तकालय" (Library). It contains fields for "Your Email" (with placeholder "example@example.com") and "Your Password" (with placeholder "Password"). There is also a "Forgot password?" link and a "Sign in" button.
- Screenshot 3: Forget Password Page** (Right): A modal window titled "Email & OTP Verification". It instructs users to enter their email to receive a one-time password (OTP) for password reset. It has fields for "Enter your email" and "Enter 6-digit OTP", along with "Verify Email" and "Verify OTP" buttons.

3. Asynchronous Task Handling with Celery: We have integrated Celery to handle time-consuming tasks, such as sending emails for password recovery and other notifications. By utilising Redis as a message broker, Celery ensures these tasks are performed asynchronously in the background, improving the overall system performance and user experience. For instance, when a user requests a password reset, an email is sent without causing delays or interruptions in the application's flow. This setup enhances the scalability and responsiveness of the platform by offloading long-running tasks from the main application thread.

4. User Registration and Data Upload

Our system allows new users to register and upload their personal data, such as profile details and preferences. This data is stored in our database and is accessible to users based on their access control level. The key highlights include:

- Access Control for Uploaded Documents: Users can upload study materials or books, which are accessible based on predefined access levels. For instance, certain documents may be restricted to specific user groups or can only be viewed by the uploader.



5. Admin/Librarian Page for User Management

We have developed a dedicated page for administrators (librarians) to manage users effectively:

- User Management: Admins can view and manage all registered users, modify their access rights, and deactivate or remove users if necessary.
- Document Oversight: Admins can also oversee uploaded documents, including managing access rights, removing inappropriate content, or archiving documents that are no longer relevant.

6. Database Integration and Load Optimization

The backend is powered by a well-structured database system, designed for scalability and optimization. The key features of our database integration include:

- Optimised Data Access: We implemented database indexing and caching mechanisms to handle the large volumes of study materials and user interactions efficiently, ensuring that the system remains responsive even under high load.
- Document and User Data Storage : All documents, user details, and interactions (like ratings and feedback) are stored securely in the database, following a well-organised schema to ensure data integrity and quick access.

7. Rating and Feedback Models for Recommendations

To provide personalised recommendations to users, we designed database models for storing ratings and feedback:

- Rating System : Users can rate study materials and books, and this data is stored in a structured model that can be queried to calculate average ratings and display them to other users.
- Feedback Model : Users can also leave feedback, which serves as additional data points for future recommendation algorithms. These feedback and rating systems will be used to provide personalised suggestions to users based on their interests and past interactions.

8. Frontend Development and CRUD Operations

Our frontend is fully functional and developed with a user-friendly interface that offers the following:

- CRUD Functionality : We have completed all basic features like adding, viewing, updating, and deleting study materials and user information. This ensures that users have full control over their uploads and can easily manage their profiles and documents.
- Intuitive User Experience : The frontend design prioritises ease of navigation, with features like a search bar for documents, a dedicated favourites section where users can save preferred books, and a dashboard to view the most popular books based on user ratings.

9. Current Progress and Future Development

We have successfully completed most of the core functionality for the system, including user authentication, document upload, access control, and user management. Some advanced features, like the recommendation system, are in progress. We plan to leverage machine learning to analyse user ratings and feedback to provide personalised book suggestions. Also try to use elastic search to search the book in large datasets.