Library Management System

Prepared by : C.D.Kasthuriarachchi

Requested by : Expernetic (Pvt) Ltd

Overview of the Project

The **Library Management System** is a web-based application designed to efficiently manage the operations of a library. It allows librarians and users to perform essential tasks such as adding, updating, searching, and deleting books, managing user authentication, and maintaining a record of available books. The system streamlines traditional manual processes and improves accessibility and organization.

Key features include:

- **Book Management** – Add, view, edit, and delete books with details like title, author, and description.
- **User Authentication** – Secure login system to restrict access to authorized users.
- **Search Functionality** – Easily search for books by title or author.
- **Admin Functionality** – User management and add , Edit or Delete Books
- **Responsive UI** – A user-friendly interface .

Technology Stack Used

- **Frontend** – React + typescript
- **Backend** – C# , .Net
- **Database** – Sqllite

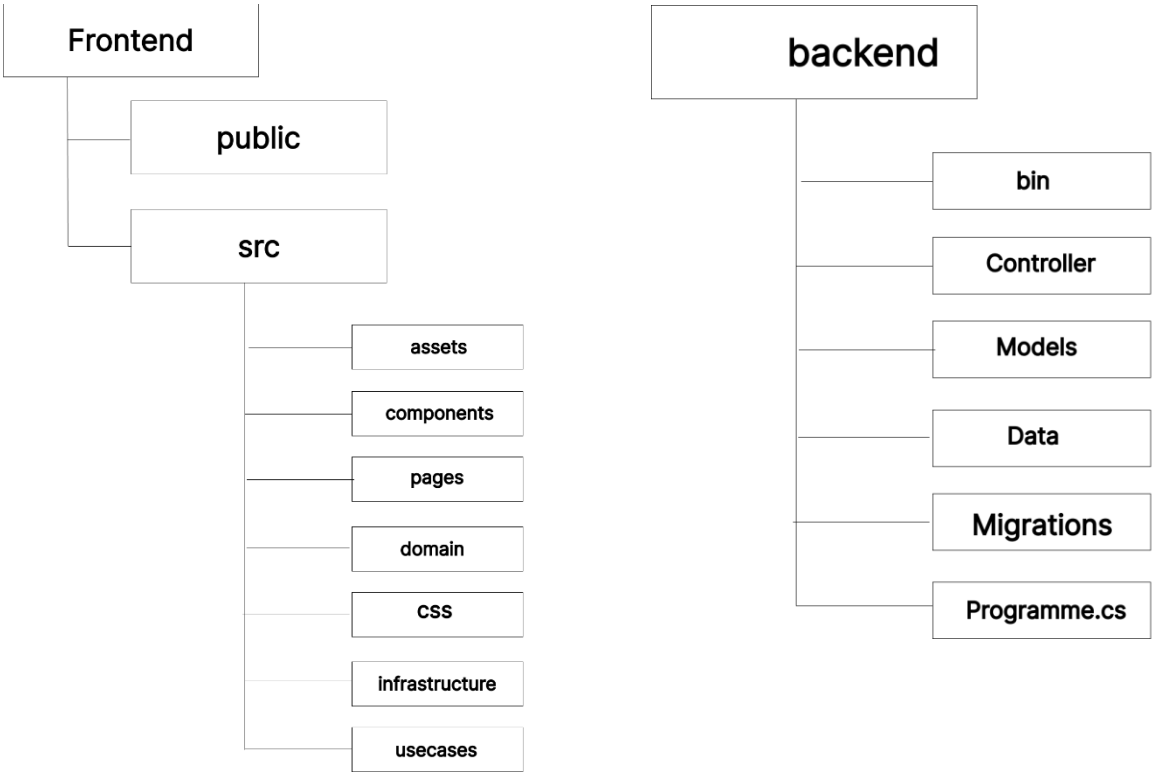
Key Features

- Add/Edit/Delete books
- Search and filter books
- User Registration and Login

Architecture or Design Pattern

This Library Management System is developed following the principles of **Clean Architecture**, which separates the application into well-defined layers with clear responsibilities. This approach enhances scalability, maintainability, and testability of the system.

File Structure



Development Process Summary

- **Requirement Analysis:**

Gathered and defined key functionalities including book management, user authentication, and search capabilities.

- **Backend Development:**

  - Designed the system architecture based on Clean Architecture principles to separate concerns.
  - Developed RESTful APIs using ASP.NET Core Web API for handling book CRUD operations and user login.
  - Used SQLite for lightweight, efficient data storage, and Entity Framework Core for database interaction.
  - Implemented security measures such as password hashing.

- **Frontend Development:**

  - Built the user interface with React, focusing on usability and responsiveness.
  - Created components for managing books, user login, and search functionality.
  - Connected frontend to backend APIs using the native **fetch API** for data requests and updates
  - Applied state management to UI updates efficiently.

- **Testing and Debugging:**

  - Conducted unit testing for backend business logic and API endpoints.
  - Performed integration testing between frontend and backend.
  - Debugged UI issues and API errors to ensure seamless interaction.

Backend Implementation
The backend is developed using **ASP.NET Core Web API** and **SQLite** for database management.

The API handles all core operations such as CRUD (Create, Read, Update, Delete) for books and user authentication.

The architecture follows Clean Architecture principles, with layers separating business logic, data access, and API controllers.

Frontend Implementation:
The frontend is built with **React**, providing a responsive and interactive user interface.

React components are organized to handle book management, search functionality, and user login.

The frontend communicates with the backend API to fetch and update data seamlessly, ensuring a smooth user experience.

Challenges Faced & Solutions

Backend Challenges

1. Working with SQLite in ASP.NET Challenge:

- Limited experience integrating SQLite with ASP.NET and configuring it with Entity Framework Core.
- Understanding how to define models, apply migrations, and interact with the database was initially difficult.

Solution:

- Researched EF Core and SQLite-specific configurations to properly set up and manage the database.
- Used EF Core migrations for smooth database schema updates.
- Implemented proper database connection handling to avoid performance issues.

Key Learning:

- Gained practical experience with SQLite and EF Core in ASP.NET, including setting up models, migrations, and data interactions.

2. Implementing a Layered Backend Architecture Challenge

- Needed to design a modular, clean, and scalable backend with proper separation of concerns.
- Initially unsure how to structure controllers, services, and repositories effectively.

Solution:

- Followed Clean Architecture principles:
  - **Controllers** handle HTTP requests and delegate to services.
  - **Services** contain business logic and communicate with repositories.
  - **DTOs** are used to safely transfer data without exposing internal models.
- Used **Dependency Injection (DI)** to maintain flexibility and modularity.

Key Learning:

- Improved understanding of layered architecture in ASP.NET.
- Learned the value of DTOs in securing and decoupling APIs.
- Strengthened skills in using DI for better testability and maintainability.

Frontend Challenges

1. Structuring Reusable Components Challenge

- Initially created large components, making the codebase harder to manage and reuse.

Solution:

- Refactored the frontend into smaller, reusable React components.
- Leveraged props and state management to maintain clean data flow between components.

Key Learning:

- Gained confidence in building scalable, readable frontend architecture using component-based design.

Additional Features

- Search and filter functionality for books by title and author to improve usability.
- User authentication with role-based access control to secure sensitive operations.
- Responsive design for accessibility across different devices.
- Admin panel to user management and book management.
- Error handling and user-friendly notifications for better interaction.

Deployment & Setup Instructions

This section provides clear steps to set up and run the Library Management System locally, including both backend and frontend setup, required dependencies, and configurations.

Backend Setup (ASP.NET Core Web API + SQLite)

Step 1: Open the Project in Visual Studio

- Launch **Visual Studio**.
- Open your ASP.NET Core Web API solution.
- Build the project to restore dependencies (NuGet will handle them automatically).

Step 2: Run the Backend

- Press **F5** or click **Start** in Visual Studio or **dotnet run** in terminal.
- The API will start (typically on https://localhost:5275) and Swagger will open in your browser.

Frontend Setup

Step 1: Install Dependencies

In your React project root, run:

npm install

This will install all required packages listed in package.json.

Step 2: Configure Backend API URL

Ensure your frontend sends API requests to the backend:

const API_BASE_URL = "https://localhost:5275";

Step 3: Start the Frontend

In your React project folder, run:

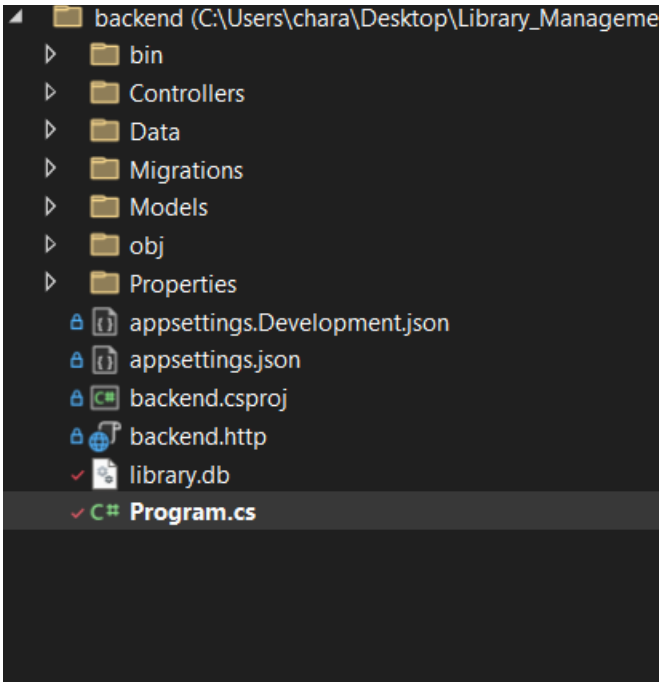
npm start

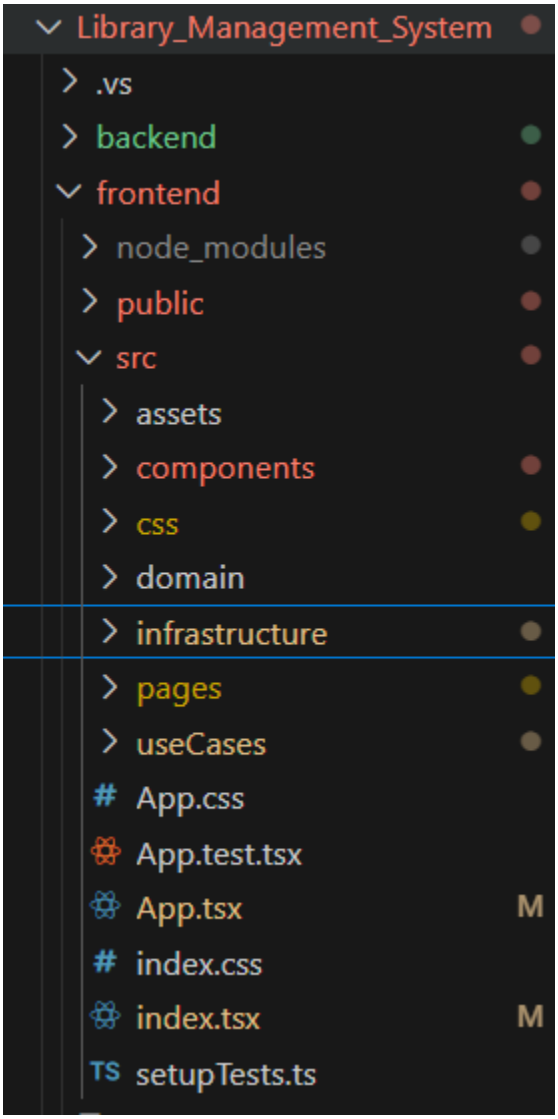
- The frontend will launch at http://localhost:3000.
- It should now be able to communicate with the backend API without any CORS issues

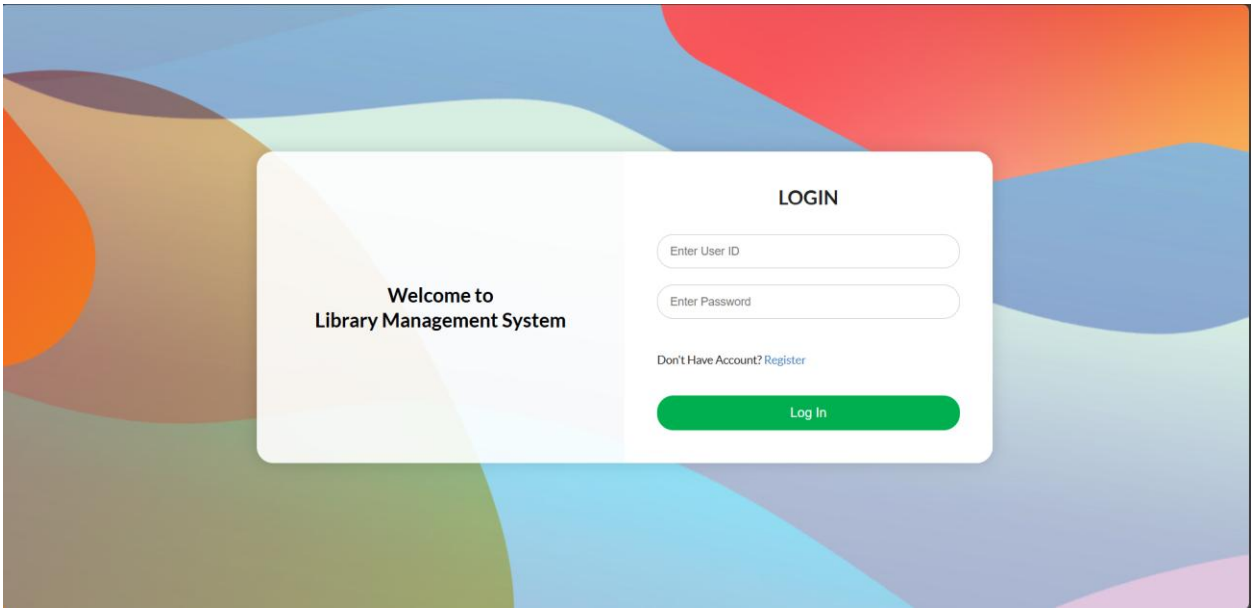
Final Notes

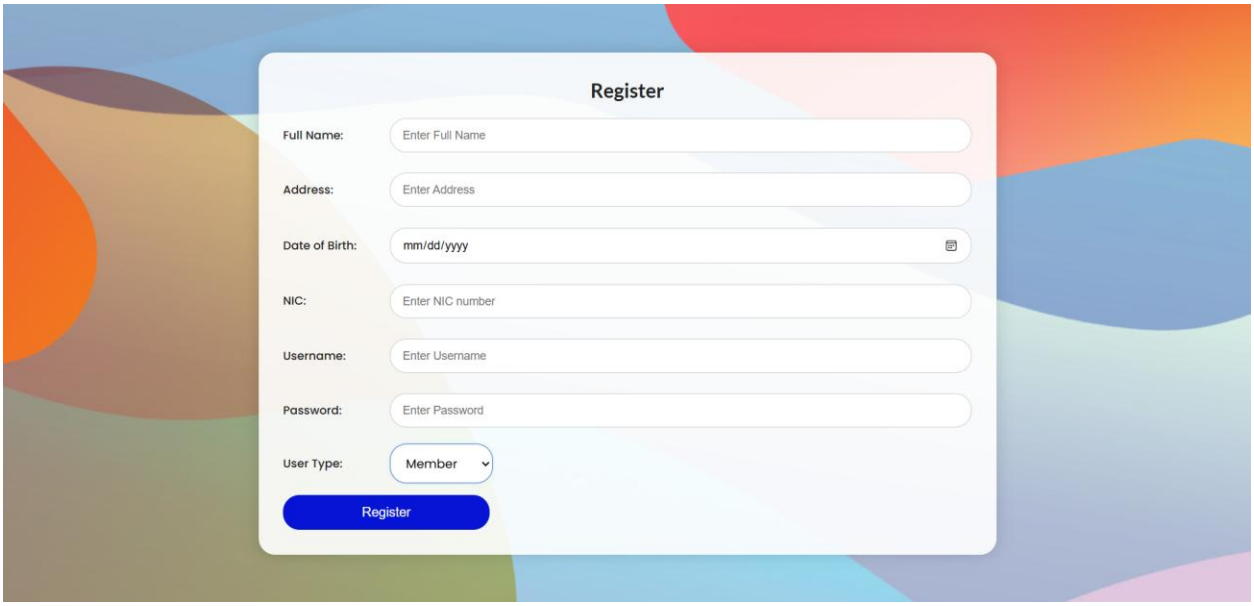
- Backend: Runs on https://localhost:5275
- Frontend: Runs on http://localhost:3000
- CORS: Fully open (AllowAnyOrigin) for development
- Swagger: Auto-launches when backend starts (useful for testing endpoints)

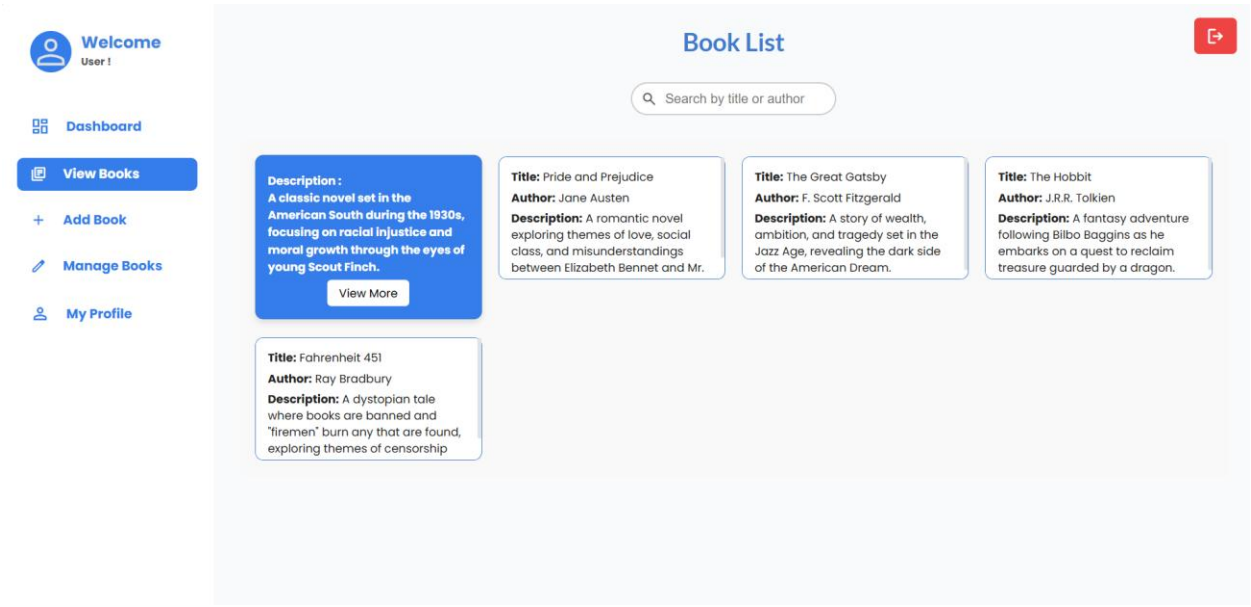
Screenshot

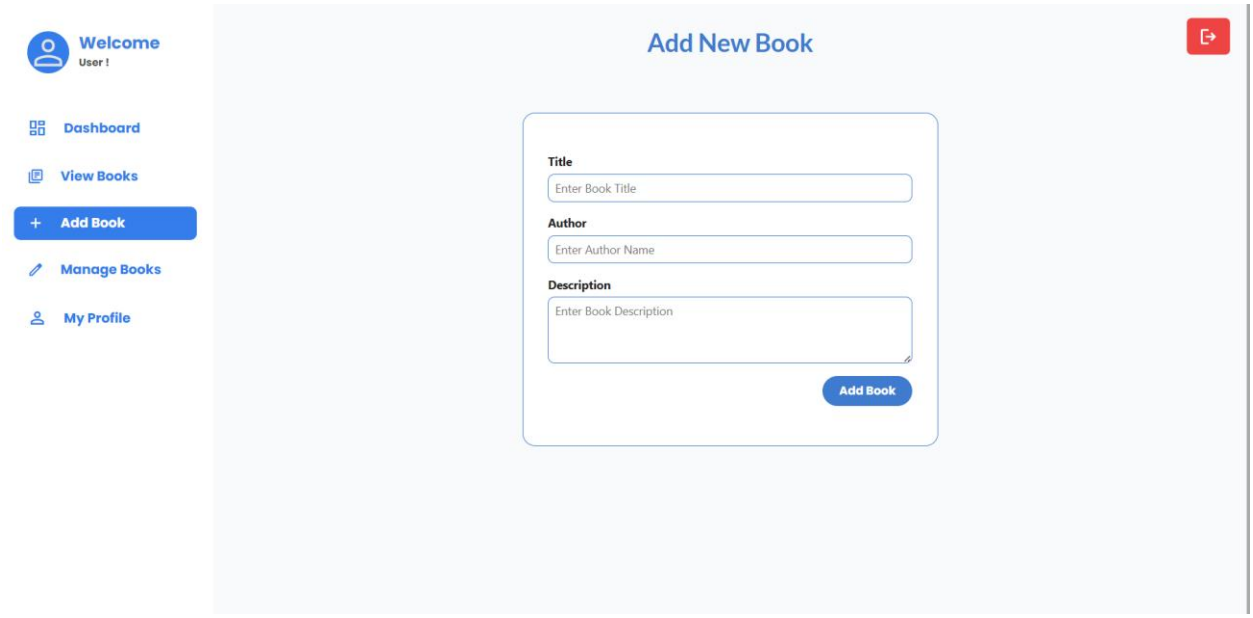


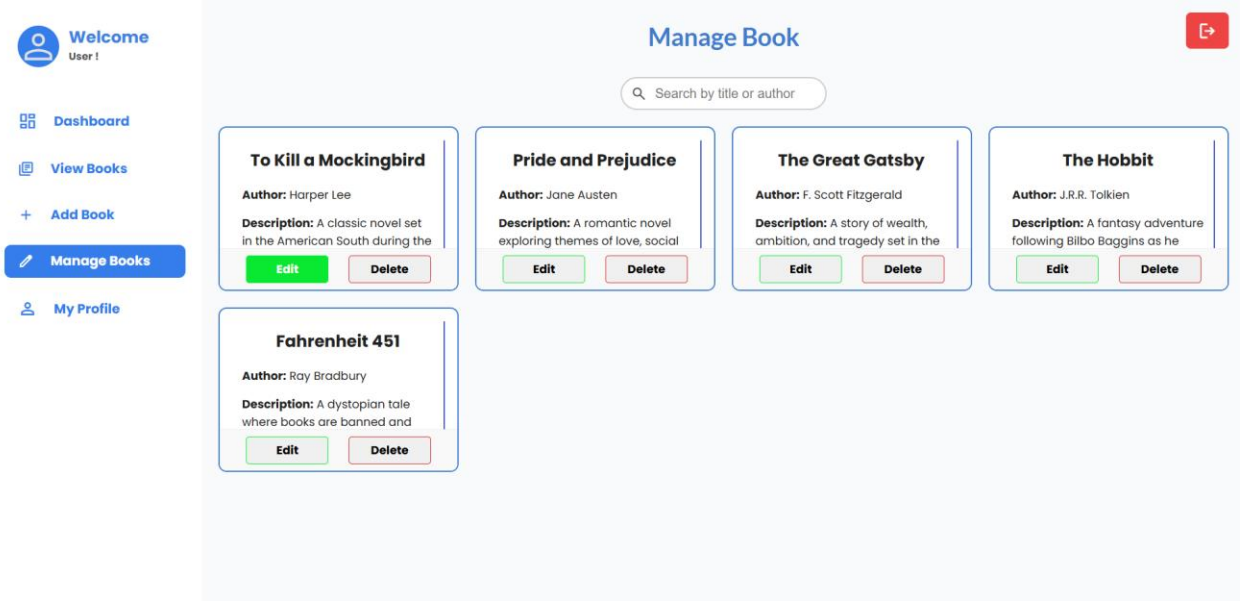


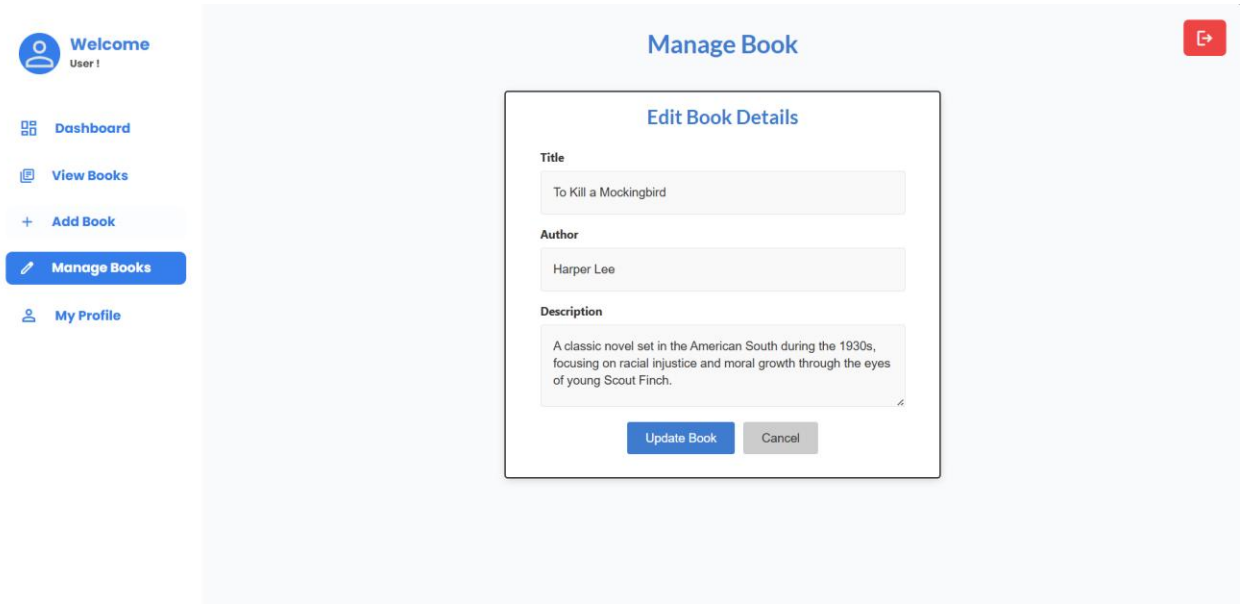

Welcome to
Library Management System
LOGIN
Enter User ID
Enter Password
Don't Have Account? Register
Log In

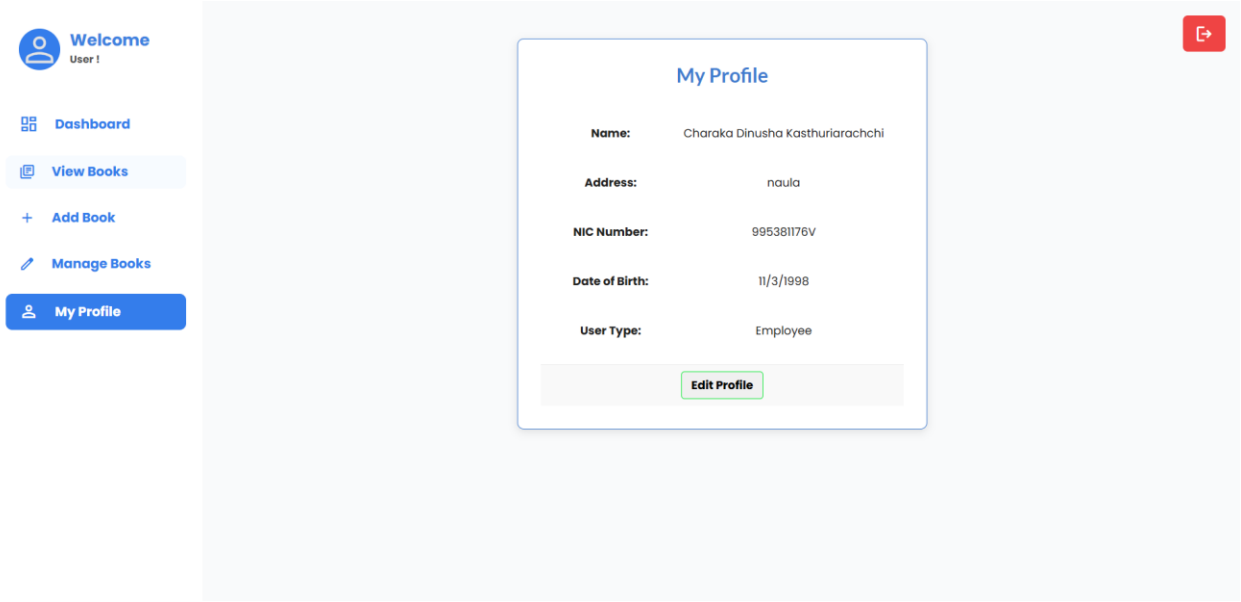

Register
Full Name:
Enter Full Name
Address:
Enter Address
Date of Birth:
mm/dd/yyyy
NIC:
Enter NIC number
Username:
Enter Username
Password:
Enter Password
User Type:
Member
Register


Welcome
User !
Dashboard
View Books
Add Book
Manage Books
My Profile
Book List
Search by title or author
Description :
A classic novel set in the American South during the 1930s, focusing on racial injustice and moral growth through the eyes of young Scout Finch.
View More
Title: Pride and Prejudice
Author: Jane Austen
Description: A romantic novel exploring themes of love, social class, and misunderstandings between Elizabeth Bennet and Mr.
Title: The Great Gatsby
Author: F. Scott Fitzgerald
Description: A story of wealth, ambition, and tragedy set in the Jazz Age, revealing the dark side of the American Dream.
Title: The Hobbit
Author: J.R.R. Tolkien
Description: A fantasy adventure following Bilbo Baggins as he embarks on a quest to reclaim treasure guarded by a dragon.
Title: Fahrenheit 451
Author: Ray Bradbury
Description: A dystopian tale where books are banned and "firemen" burn any that are found, exploring themes of censorship

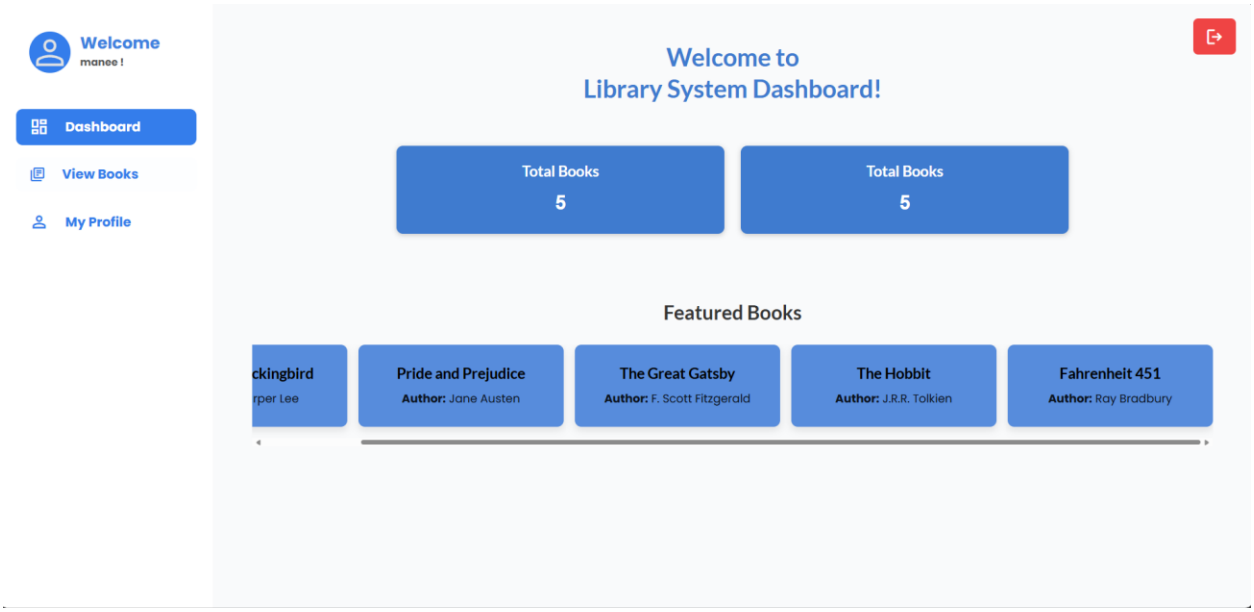

Welcome
User !
Dashboard
View Books
Add Book
Manage Books
My Profile
Add New Book
Title
Enter Book Title
Author
Enter Author Name
Description
Enter Book Description
Add Book


Welcome
User !
Dashboard
View Books
Add Book
Manage Books
My Profile
Manage Book
Search by title or author
To Kill a Mockingbird
Author: Harper Lee
Description: A classic novel set in the American South during the
Edit
Delete
Pride and Prejudice
Author: Jane Austen
Description: A romantic novel exploring themes of love, social
Edit
Delete
The Great Gatsby
Author: F. Scott Fitzgerald
Description: A story of wealth, ambition, and tragedy set in the
Edit
Delete
The Hobbit
Author: J.R.R. Tolkien
Description: A fantasy adventure following Bilbo Baggins as he
Edit
Delete
Fahrenheit 451
Author: Ray Bradbury
Description: A dystopian tale where books are banned and
Edit
Delete

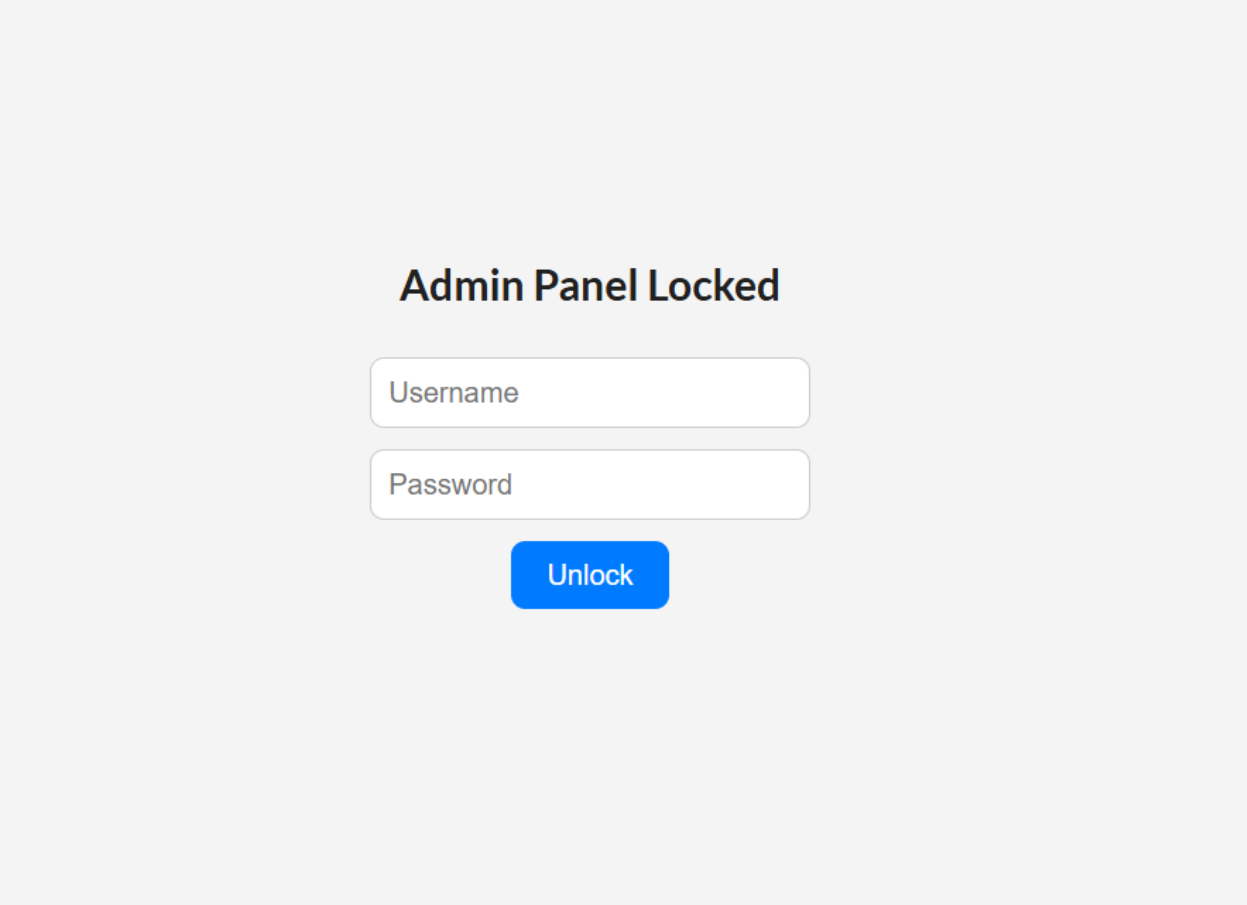

Welcome
User !
Dashboard
View Books
Add Book
Manage Books
My Profile
Manage Book
Edit Book Details
Title
To Kill a Mockingbird
Author
Harper Lee
Description
A classic novel set in the American South during the 1930s, focusing on racial injustice and moral growth through the eyes of young Scout Finch.
Update Book
Cancel

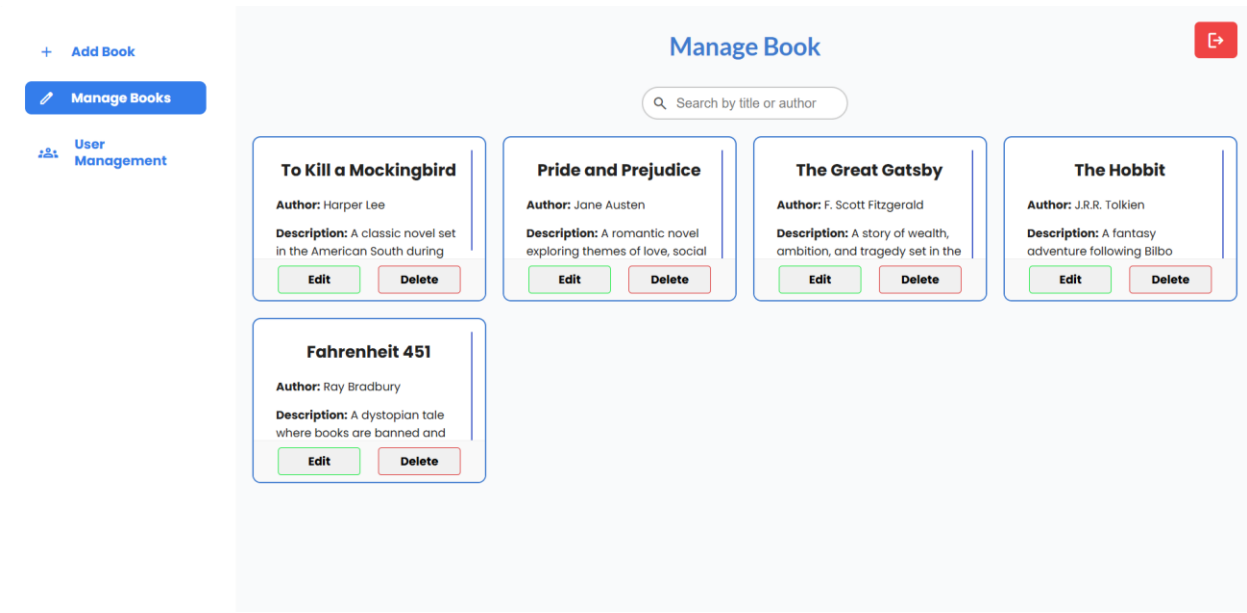

Welcome
User !
Dashboard
View Books
Add Book
Manage Books
My Profile
My Profile
Name:
Charaka Dinusha Kasthuriarachchi
Address:
naula
NIC Number:
995381176V
Date of Birth:
11/3/1998
User Type:
Employee
Edit Profile

Member View


Welcome
manee !
Dashboard
View Books
My Profile
Welcome to
Library System Dashboard!
Total Books
5
Total Books
5
Featured Books
ckingbird
rper Lee
Pride and Prejudice
Author: Jane Austen
The Great Gatsby
Author: F. Scott Fitzgerald
The Hobbit
Author: J.R.R. Tolkien
Fahrenheit 451
Author: Ray Bradbury

Admin Panel


Admin Panel Locked
Username
Password
Unlock


Add Book
Manage Books
User Management
Manage Book
Search by title or author
To Kill a Mockingbird
Author: Harper Lee
Description: A classic novel set in the American South during
Edit
Delete
Pride and Prejudice
Author: Jane Austen
Description: A romantic novel exploring themes of love, social
Edit
Delete
The Great Gatsby
Author: F. Scott Fitzgerald
Description: A story of wealth, ambition, and tragedy set in the
Edit
Delete
The Hobbit
Author: J.R.R. Tolkien
Description: A fantasy adventure following Bilbo
Edit
Delete
Fahrenheit 451
Author: Ray Bradbury
Description: A dystopian tale where books are banned and
Edit
Delete