MODERN APPLICATION DEVELOPMENT - PROJECT

LIBRARY MANAGEMENT SYSTEM

Name-> Anirudh Pabbaraju

Email -> 23f2002166@ds.study.iitm.ac.in

Roll No -> 23f2002166

I am Anirudh currently pursuing my bsc degree from IITM and I'm in diploma first term and also pursuing BTECH from NITW currently in 2nd year I'm excited to submit my mad-1 project

Description

This is a project about Online Library Management. There will be one or more admins and many users for the web dev app.

There is many Sections and many books in those sections for the users to read or purchase .

The admin can perform CRUD Operations on Section and Books .

Technologies used

- Flask as it helps me build web applications in Python, making it easier to handle user requests, manage routing, and create web pages.
- Flask-SQLAlchemy to work with databases in my Flask application.
- Jinja2 as the templating engine to generate dynamic HTML content. It allows me to combine Python code with HTML templates, making it easier to display data from my application.
- Bootstrap for styling tables , forms and buttons .and CSS for styling

* Flask_restful for implementing API stuff for section and books

* Matplotlib for plots, histograms ,pie chart and bar char for stats section

DB Schema Design

- **User Table**: Contains user information such as user first and last name, password, user email. Each user has a unique ID and can be associated with book requests.
- **Section Table**: Represents different sections in the library. Each section has a unique ID, name, creation date, and description.

* **Librarian Table** : the information of one or more admins available here (librarian first and last name ,email and password and unique id)

- **Book Table**: Stores information about books available in the library. Includes details like book name, content, authors, date added, and the section it belongs to.

Each book has a unique ISBN_No number and there are multiple copies of books available

. • **Book_Issue Table**: Records book issues to users, indicating which books are issued, when they were issued, return dates, due_dates , request dates and ISBN_no of book,user_id

* **Keeps_track_of** -> has librarian_id and user_id of those users monitored by that librarian

* **Manage** -> this has librarian_id and section_id of those sections managed by that particular librarian

* **Review**s -> has user_id,isbn and review and rating for a particular book by a user

* **Roles** -> has user_id and type of roles and quota for every role

Architecture and Features

app.py file contains the setup for my Flask application, including creating the Flask app object,

setting up the database connection, and importing necessary files like user_controller.py ,librarian_controller.py, section_controller.py and book_controller.py

models.py where we have all models possible all database tables

4 controller.py for user,librarian and section and book in section and book we have CRUD controllers 4+4 and in user and librarian other thing like logging in requesting for a book

Returning a book , granting permission ,rejecting permission , revoking a book etc etc

Api.py we have api linked with section and book

Basic routes for user and admin :

Register ,login ,dashboard,

User Routes -.> 1) Searching for a book/section by name or author name ,

2)Route for seeing their books ,all avalaible books

3) stats to see number of books taken in each section etc

4) route for requesting a book

5) withdrawing a request for a book ,

6) returning a book

7) reissuing a book

8) while returning can give feedback

Librarian Routes 1) View Requests sent by users

2) Monitoring all users associated with him

3) Granting permission for books

4) Rejecting permissions

5) Revoking books before due date (can do it)

6) stats section to see number of book in each section a pie chart

Section and Book Routes : CRUD applications , add_section edit_section , delete-section view_section , add_book, edit_book,delete_book,view book

Video -> https://www.loom.com/share/0d7713eea45b4af090d15cba1d80c538?sid=bfc29b20-4d02-4516-9ce8-ebf5ec9e28d2