Name - Shobhit Shukla
Roll No.- 21f1006883
Email Id- 21f1006883@ds.study.iitm.ac.in
I am currently a diploma-level student at IITM and simultaneously pursuing MCA from CSJM University, Kanpur.

Description

I want to integrate a proper payment gateway for purchasing books, but due to my current lack of knowledge about it, I have skipped that part for now. However, I plan to implement a proper payment gateway in the future as part of the MAD-2 project.

Technologies used

- Flask: Used for application code.
- Jinja2 Template and Bootstrap: Used for HTML generation and styling.
- Flask-SQLAlchemy: Used for the database.
- Matplotlib: Used for comprehensive visualization in Python.

DB Schema Design

User Table:

- id: Primary key, auto-incremented integer.
- name: String, up to 100 characters, cannot be null.
- phone: String, up to 20 characters, cannot be null, unique.
- email: String, up to 50 characters, cannot be null, unique.
- password: String, up to 50 characters, cannot be null.
- role: String, up to 20 characters, cannot be null, default value is 'customer'.

Section Table:

- id: Primary key, auto-incremented integer.
- section_name: String, up to 100 characters, cannot be null.
- date_created: DateTime, cannot be null, default value is the current UTC date and time.
- description: Text, cannot be null.
- image: String, up to 255 characters, cannot be null.

Book Table:

- id: Primary key, auto-incremented integer.
- title: String, up to 100 characters, cannot be null.
- author: String, up to 100 characters, cannot be null.
- content: Text, cannot be null.
- image: String, up to 200 characters, cannot be null.
- pdf: String, up to 200 characters, cannot be null.
- section_id: Integer, foreign key referencing the id column of the Section table, cannot be null.
- date_created: DateTime, cannot be null, default value is the current UTC date and time.
- price_per_day: Float, cannot be null.

PurchaseRequest Table:

- id: Primary key, auto-incremented integer.
- user_id: Integer, foreign key referencing the id column of the User table, cannot be null.
- book_id: Integer, foreign key referencing the id column of the Book table, cannot be null.
- duration: Integer, cannot be null.
- price: Float, cannot be null.
- status: String, up to 20 characters, cannot be null, default value is 'pending'.
- request_datetime: DateTime, cannot be null, default value is the current UTC date and time.

Reasons behind the design:

- User table: Stores information about users such as name, contact details, and role. The role column allows for distinguishing between customers and librarians.
- Section table: Represents different sections in the library, each with a name, description, and image.
- Book table: Contains details about books, including title, author, content, image, and PDF link. The section_id column establishes a relationship with the Section table, allowing books to be categorized by section.
- PurchaseRequest table: Tracks purchase requests made by users for books. Includes information about the user, the book requested, duration, price, status, and request datetime.

API Design

I have not implemented API because of lack of understanding about API but I implemented JavaScript Validation in user Registration page.

Architecture and Features

The project is organized following the Model-View-Controller (MVC) architecture. Controllers, located in routes.py, manage the application logic and handle incoming HTTP requests. Templates, stored in the templates directory, define the presentation layer and are rendered dynamically by Flask based on controller actions.

As for the features, the project includes standard functionalities such as user authentication, user registration, and session management. Additionally, it offers features like searching for sections and books, displaying search results, viewing book details, and purchasing books.

Video

https://drive.google.com/file/d/1oYVoC2H804K1xHF6okWel9qX1aIy-C6y/view?usp=drive_link