Problem Approach

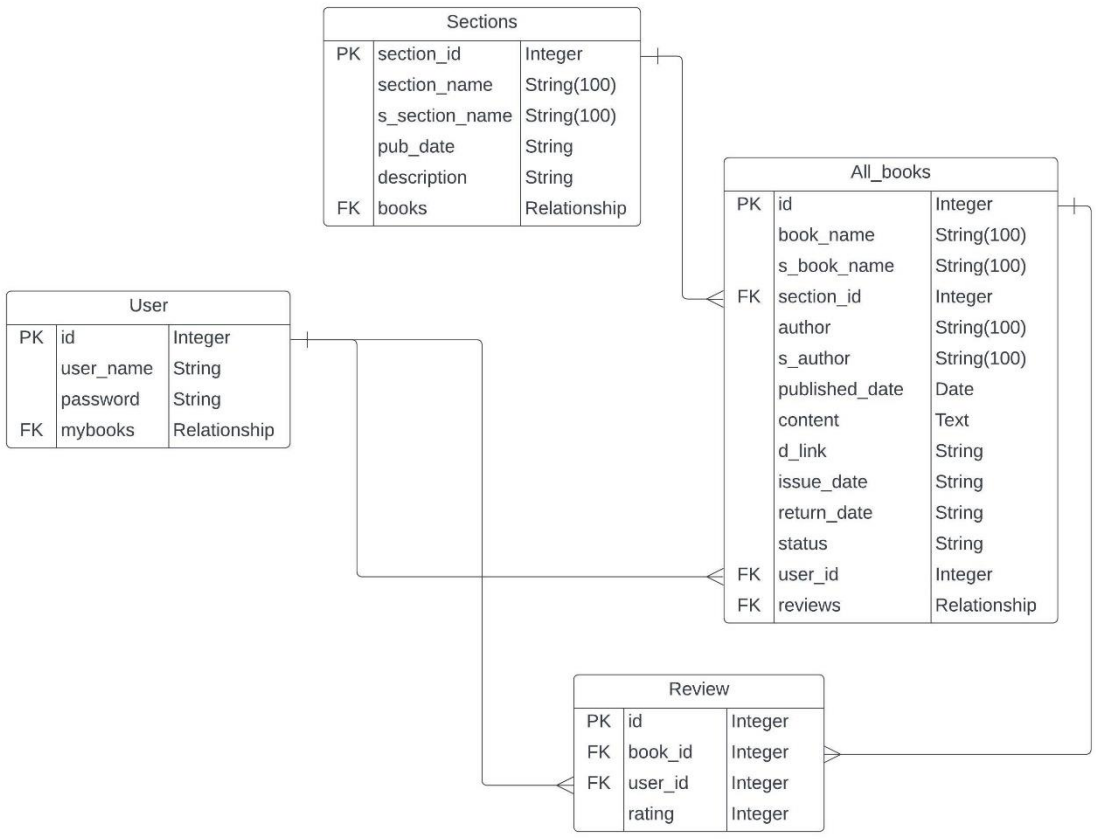
- The project began with the creation of HTML templates to define the structure and layout of the web application with the help of bootstrap.
- Once the HTML templates were in place, next the SQLite database models were defined to represent the data structure and relationships within the application.
- The next step involved writing the Flask for the application code for the CRUD operations of the application.
- Additional features were added and also the testing of the application for errors bring the completion of the project.

About The Library Management System

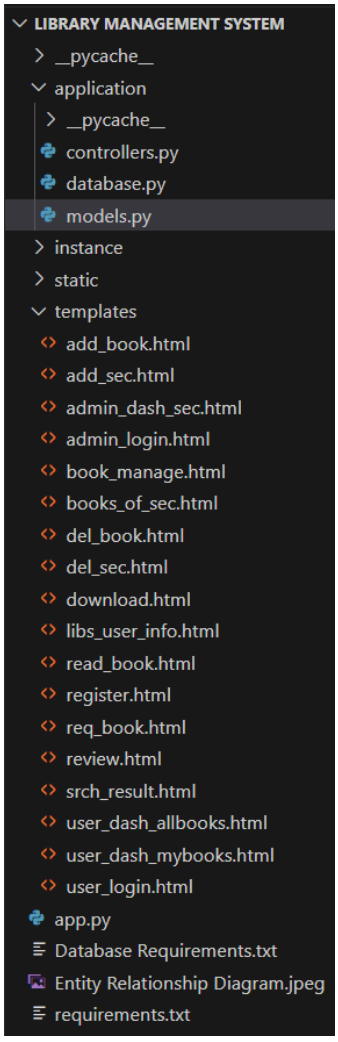
- It is a hosted web application using flask application code.
- Uses Jinja2 templates and Bootstrap for HTML generation and styling.
- SQLite database is used for storing the data.
- The application is a multi-user app (one admin/librarian and multiple general users) where the user can request, read and return book.
- Librarian/Admin can add new sections/e-books, issue/revoke access for a book.

Database Model (Entity Relationship Diagram) of the project

- Below ER diagram specify the database models and their relations.



File Management of the Project



- The ***controllers.py*** contains all the endpoints of the web application, and contains most of the back-end code. Rest of it are distributed between ***models.py*** (defines the tables for SQLAlchemy classes), ***database.py*** (hooks up the database to models.py).
- The ***static*** folder contains the CSS stylesheets which are used in the design of the webpages of the application.
- Application may store instance-specific data such as local databases, logs, or other persistent data files within the ***instance*** folder.
- The ***templates*** folder contains HTML files that serve as templates for rendering web pages as shown above in the attached picture.

Submitted By :-

Aditya Ambati

Email:- 21f3000274@ds.study.iitm.ac.in

For a walkthrough of my project check this video.