## Author

Saravanan
23F2002620
23f2002620@ds.study.iitm.ac.in
A Design Thinker and Dual-Degree Student with Engineering background.

## Description

This project is a multi-user application designed as an exam preparation platform for multiple courses. It provides users with the ability to attend quizzes for different chapters across various subjects.

## Technologies used

- Flask – This framework is a lightweight web framework for building web applications. It provides flexibility and simplicity for creating scalable and maintainable applications. This framework is easy to integrate with other extensions like Flask-SQLAlchemy.
- Werkzeug - It is a utility library that works behind the scenes in Flask to manage HTTP requests and responses. It helps with routing, debugging, and error handling. Essentially, it acts as a toolkit that makes building web applications smoother and more efficient.
- io Module – This module in Python is used for handling input/output operations, such as reading/writing files or streams. It used for managing file uploading/downloading within the application. It facilitates dynamic content generation (e.g., exporting quiz results).
- Matplotlib – This is a Python library for creating static, animated, and interactive visualizations. It is used to generate charts and graphs for performance analytics. It is in implementing Summary charts.
- Datetime - The datetime module is used to handle date and time operations. It is useful for timestamping quiz attempts, tracking user activity, or scheduling quizzes.
- Flask-SQLAlchemy - It is an extension of Flask that integrates SQLAlchemy, a powerful Object Relational Mapper (ORM), with Flask applications. It simplifies database interactions by allowing developers to work with Python objects instead of raw SQL queries. It provides tools to define models, manage relationships between tables (e.g., users, quizzes, chapters), and query data efficiently.

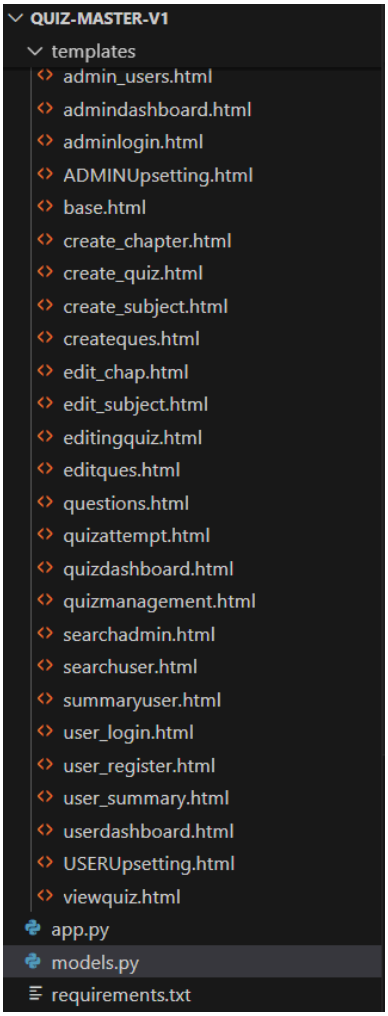
## DB Schema Design

Here’s a structured database schema design for your exam preparation platform, optimized for scalability, security, and efficient querying:


The above database of six tables: User, Subject, Chapter, Quiz, Questions, and Scores. The User table stores user details such as username, password, role, and other personal information. The Subject table represents subjects, each of which can have multiple chapters stored in the Chapter table. Chapters are linked to subjects through a foreign key (Subject_id). Similarly, quizzes are associated with chapters via a foreign key (Chapter_id) in the Quiz table, and each quiz can have multiple questions stored in the Questions table and multiple scores recorded in the Scores table. Scores link quizzes and users through foreign keys (Quiz_id and User_id). Relationships between tables are defined using SQLAlchemy's relationship feature, enabling efficient navigation across related data. This schema ensures referential integrity and supports functionalities like managing users, subjects, chapters, quizzes, questions, and tracking scores for attempted quizzes.

## Architecture and Features

The project is organized into a structured directory that separates the core components of the application for maintainability and scalability. The templates folder contains all the HTML files used for rendering views in the application. These templates are organized based on functionality, such as user management (user_login.html, user_register.html), quiz creation and editing (create_quiz.html, editquiz.html), and dashboards for both users and admins (userdashboard.html, admindashboard.html). The app.py file serves as the main entry point of the application, likely containing route definitions and controllers that handle HTTP requests and responses. The models.py file defines the database schema using Flask-SQLAlchemy, mapping Python classes to database tables. Lastly, the requirements.txt file lists all the dependencies required to run the project, making it easy to set up the environment.




## Video

https://drive.google.com/file/d/1a2uSyIJhGFZub0x5M5AP0I6erawk0aN3/view?usp=sharing