MAD1 -Project Report

Author
Name:- Anirudh Singh Tomar
Roll no:- 23f3003917
Email Id: 23f3003917@ds.study.iitm.ac.in

I am a motivated and enthusiastic student with a deep interest in software development and modern application design. I enjoy building web-based applications that effectively integrate frontend and backend technologies.

Description :- I have designed **Quiz Master** a multi-user exam preparation site using **Flask, Jinja2, SQLite, HTML, CSS, and Bootstrap**. The **Admin** manages subjects, chapters, quizzes, and questions, while **Users** take quizzes and track scores.

Technologies used

Backend:
Flask: Lightweight Python web framework for building the application.
Flask-SQLAlchemy: ORM for database management with SQLite.
Flask-RESTful: Helps create RESTful APIs for future expansion.
Flask-Login: Manages user authentication and sessions.

Frontend:
Jinja2: Template engine for rendering dynamic content.
HTML, CSS, Bootstrap: Frontend styling and responsive design.

Database:
SQLite: Lightweight database for storing users, quizzes, and scores.

Purpose of These Technologies:
Flask & Extensions: Simplify backend logic, authentication, and database management.
Jinja2 & HTML/CSS: Enable dynamic rendering and structured UI.
Bootstrap: Ensures a responsive and modern-looking admin panel.
SQLite: Provides a simple, file-based database solution for easy deployment

DB Schema Design :- **Subject table**

Column Name	Data Type	Constraints	Description
id	integer	Primary key	Unique subject ID
Name	text	Unique, not null	name
Description	text		description

Chapter table

Column Name	Data Type	Constraints	Description
id	integer	PRIMARY KEY	Unique chapter id
name	text	NOT NULL	Chapter name
Subject_id	integer	FOREIGN KEY	Associated subject

Quizzes Table

Column Name	Data Type	Constraints	Description
id	integer	Primary key	Quiz id
Title	text		Quiz title
description	text	Not null	Quiz detaails

Question Table:

Column Name	Data Type	Constraints
Quiz text	string	Not null
Option A	string	Not null
Option B	string	Not null
Option C	string	Not null
Option D	string	Not null

Score table:-

Column Name	Data Type	Constraints
Id	Integer	Not null
score	Integer	Not null
Time stamp	Date time	Not null

Why this design?
- **Normalization** prevents redundancy.
- **Foreign Keys & Cascade Deletion** ensure relational integrity.
- **User Tracking** enables score analysis and performance monitoring

API Design :- The API, built with Flask-RESTful, manages authentication, subjects, quizzes, and user attempts. It uses Flask-SQLAlchemy for the database, Marshmallow for validation, and JWT for security, ensuring a structured and scalable system.

Architecture and Features:- The project follows the MVC (Model-View-Controller) architecture. The controllers (routes and API endpoints) are managed in the routes module, handling user authentication, quiz management, and admin functionalities. The models are defined using Flask-SQLAlchemy, structuring subjects, quizzes, users, and scores. The templates, styled with Bootstrap, are located in the templates folder, using Jinja2 for dynamic rendering. The static files (CSS, JavaScript) are in the static directory. Admin and user dashboards, along with API integrations, ensure smooth quiz management and data visualization**.**

Features Implemented
The project includes core features such as user authentication (registration & login) using Flask-Login, quiz management (subjects, chapters, quizzes, and questions) handled through Flask-SQLAlchemy models and Jinja2 templates, and score tracking with database integration.
Admin features include subject, chapter, quiz, and question management with CRUD operations, implemented via Flask routes and form handling. Users can take quizzes, and their scores are stored and displayed through summary charts using Chart.js.
Additional features include an Admin Dashboard with analytics, a search functionality for easy navigation, and a RESTful API for quiz data, designed using Flask-RESTful.

Video Link -

https://drive.google.com/file/d/1eqGt65wlgDdO3yXwJPOUksJJFBbB4B8f/view?usp=sharing