**PROJECT REPORT – QUIZ MASTER V1**

**AUTHOR**

My name is Murari Vijay, Roll No. DS24f2005340, email:[24f2005340@ds.study.iitm.ac.in](mailto:24f2005340@ds.study.iitm.ac.in). I am a 3rd year B.Tech Biotechnology Student in Rajalakshmi Engineering College.

**DECRIPTION**

Develop a Quiz Application that includes user registration, separate admin and user login. The admin can add, delete, and update subjects, chapters, quizzes, and questions. Users can attempt quizzes, view scores, and see the leaderboard.

**TECHNOLOGIES USED**

Flask – This is the python-based server

Jinja2 – Used to render HTML templates

SQLite – Used as Database for the application

SQLAlchemy – Facilitates communication between python programs and databases.

**DB DESIGN SCHEMA**

The Database has 6 Tables:

* Users (uid Integer PRIMARY\_KEY, name STRING, email STRING, password STRING, dob DATE)
* Subjects (sid INTEGER PRIMARY\_KEY, sname STRING)
* Chapter (cid INTEGER PRIMARY\_KEY, cname STRING, csid INTEGER FOREIGN\_KEY)
* Quiz (zid INTEGER PRIMARY\_KEY, zname STRING, zdate DATE, zdur INTEGER, zsid INTEGER FOREIGN\_KEY)
* Question (qid INTEGER PRIMARY\_KEY, qtitle STRING, qquestion STRING, qopt1 STRING, qopt2 STRING, qopt3 STRING, qopt4 STRING, qcorrect INTEGER, qzid INTEGER FOREIGN\_KEY, qcid INTEGER FOREIGN\_KEY)
* Score (rid INTEGER PRIMARY\_KEY, rtot INTEGER, rtstp DATE, rqid INTEGER FOREIGN\_KEY, rsid INTEGER FOREIGN\_KEY)
* Users table stores the list of users.
* Subjects table stores the list of subjects.
* Chapters table stores the list of chapters.
* Quiz table stores the list of quizzes.
* Question table stores the list of questions.
* Score table stores the list of scores.

**API DESIGN**

APIs handle all website functionalities and provide data for each page. SQLAlchemy was used to communicate with the SQLite3 database.

Created APIs to:

* Register a user and sign in a user, as well as logout
* Sign in as Admin, as well as logout
* Add, Delete, Edit and Update quizzes, chapters, subjects and questions (3 separate APIs with separate routes for all parameters)
* Fetch all Scores from the quizzes attempted the user logged in along with the details according to timestamps as well.
* Fetch list of users registered
* Serve all the web pages

**ARCHITECTURE AND FEATURES**

The whole project is inside the ‘QuizMaster Code’ folder.

Within this folder, the ‘templates’ folder contains all the html web page files.

‘project.db’ is the database file which is inside the instances folder.

‘static’ folder contains all the css files and required pictures.

‘models.py’ contains all the models of tables that is used in the database.

‘app.py’ contains all the controllers and implements the functionalities of the website.

All Features:

* User signup and login with validation (Signup only works if username doesn’t already exist, returns alert if already exists)
* Admin login with validation.
* Subjects, Chapters, Quizzes and Questions Management (Create, Edit, Delete)
* User’s Scores (Fetches scores according to marks, highest mark shows up)
* Summarization of all the user’s score in Admin Summary.
* Comparison Scores with other users in the User’s Leaderboard.

**VIDEO LINK**

https://drive.google.com/file/d/1Q0g6dLL2l3beYz4rheA7XLhVHnsNSFdi/view?usp=sharing