

Author

Name: GEORGE CHERIAN THALIATH

Roll no: 22F1001875

Email: 22f1001875@ds.study.iitm.ac.in

I'm a B.Tech graduate in Computer Science with a passion for learning. Committed to continuous growth and excellence

Problem Statement

Quiz Master is a multi-user app (one requires an administrator and other users) that acts as an exam preparation site for multiple courses

Description

Quiz Master is a multi-user quiz preparation platform where an admin (quiz master) manages subjects, chapters, quizzes, and questions, while registered users can attempt quizzes and track scores. The platform is built using Flask, SQLite, and Bootstrap.

Technologies used

Flask: provides a lightweight and flexible web framework.

Jinja2: enables dynamic HTML content rendering.

SQLite: simple and efficient database choice for local storage

Flask-SQLAlchemy: provides an ORM to simplify database operations.

Chart.js: allows visualization of quiz performance

DB Schema Design

1. Users Table

- **id:** (Integer, Primary Key, Auto-Increment) – Unique identifier for each user.
- **is_superuser:** (Integer, Default: 0, Not Null) – Identifies whether the user is an admin (1) or a regular user (0).
- **mail:** (String, Unique, Not Null) – User's email for login.
- **pwd:** (String, Not Null) – Encrypted password for authentication.
- **f_name:** (String, Not Null) – User's full name.
- **qual:** (String) – User's qualification.
- **dob:** (Date) – User's date of birth.

2. Subject Table

- **id:** (Integer, Primary Key, Auto-Increment) – Unique identifier for each subject.
- **name:** (String, Unique, Not Null) – Name of the subject.
- **description:** (Text, Nullable) – Additional details about the subject.

3. Chapter Table

- **id:** (Integer, Primary Key, Auto-Increment) – Unique identifier for each chapter.
- **name:** (String, Not Null) – Name of the chapter.
- **s_id:** (Integer, Foreign Key → Subject.id, CASCADE delete) – Associates the chapter with a subject.

4. Quiz Table

- **id:** (Integer, Primary Key, Auto-Increment) – Unique identifier for each quiz.
- **date:** (Date, Not Null) – The scheduled date of the quiz.
- **time_duration:** (Integer, Not Null) – Time duration of the quiz (in minutes).
- **remarks:** (Text) – Optional remarks or instructions for the quiz.
- **c_id:** (Integer, Foreign Key → Chapter.id, CASCADE delete) – Associates the quiz with a chapter.

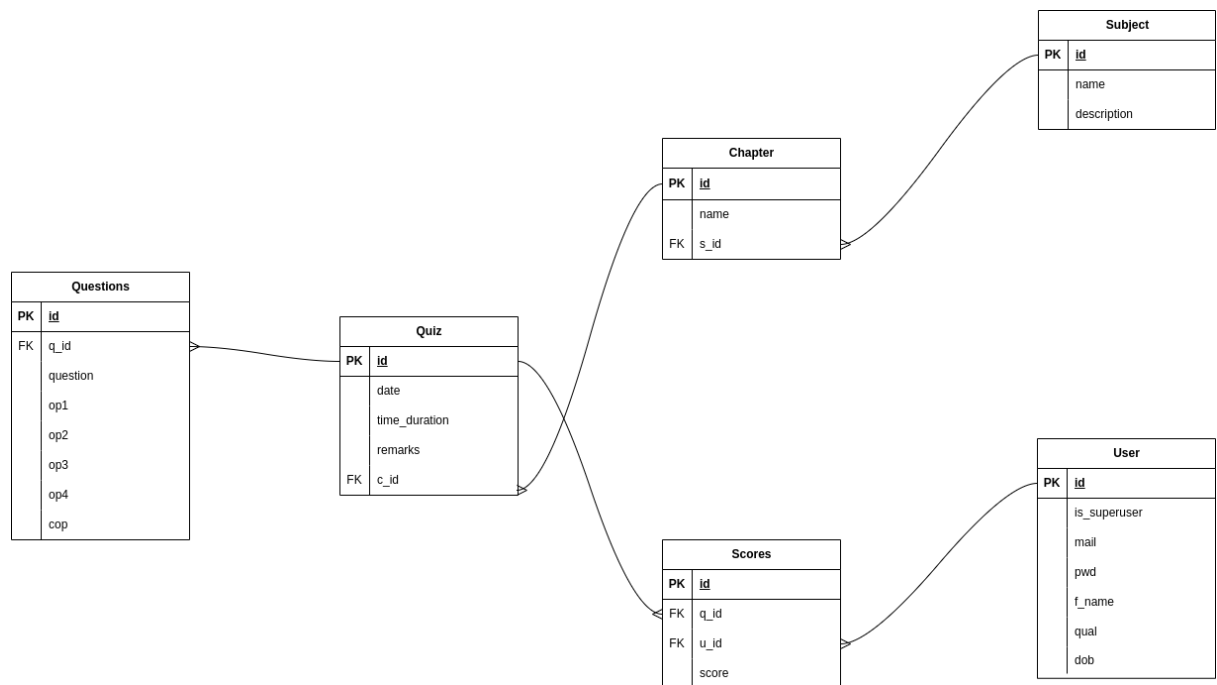
5. Questions Table

- **id:** (Integer, Primary Key, Auto-Increment) – Unique identifier for each question.
- **q_id:** (Integer, Foreign Key → Quiz.id, CASCADE delete) – Associates the question with a quiz.
- **question:** (Text, Not Null) – The question statement.
- **op1, op2, op3, op4:** (Text, Not Null) – Multiple-choice answer options.
- **cop:** (Integer, Not Null) – The correct option (1-4).

6. Scores Table

- **id:** (Integer, Primary Key, Auto-Increment) – Unique identifier for each quiz attempt.
- **q_id:** (Integer, Foreign Key → Quiz.id, CASCADE delete) – Associates the score with a quiz.
- **u_id:** (Integer, Foreign Key → Users.id, CASCADE delete) – Associates the score with a user.
- **score:** (Integer, Not Null) – The total score obtained by the user.

ER Diagram



Architecture and Features

Project Organization:

The root folder is **quiz_master_22f1001875**. It contains `app.py` and the following folders: `controllers`, `instance`, `models`, `static`, and `templates`.

- The **controllers** folder contains `controllers.py`, which includes all the project's controllers.
- The **instance** folder contains the database.
- The **models** folder contains `models.py`, which defines all Flask models.
- The **static** folder includes an `images` subfolder that stores all project-related images.
- The **templates** folder contains all the HTML files.

Implemented features:

- Designed a **database schema** for users, subjects, quizzes, and scores.
- Implemented **user registration** and **admin login** functionality.
- Built an **Admin Dashboard** with **CRUD operations** for subjects, chapters, quizzes, and questions.
- Developed a **User Dashboard** with quiz attempt functionality and a timer. *(Note: The timer is implemented in the backend but not displayed on the front end.)*
- Implemented **score tracking** and **quiz result display**.
- Added **search functionality** for both Admin and Users.
- Enabled **quiz scheduling** with time duration settings.
- Integrated **summary charts** for quiz statistics and user performance.

- Enhanced the **UI with Bootstrap styling** and **form validation**.

Video

<https://drive.google.com/file/d/19UbvBxXLPeYoXNgV58mfbe6ojlbtOlnS/view?usp=sharing>