Flask-Based Quiz Management System
Quizzer

Author

Name: Niranjan Suresh
Roll Number: 23f3002251
Email: 23f3002251@ds.study.iitm.ac.in

About me: Hello my name is Niranjan Suresh. I am an aspiring software developer with a strong passion for building applications. My interests lie in full-stack web development. I am also deeply fascinated by data science and I look forward to exploring machine learning and analytics in future projects. I'm also passionate about music and badminton.

Description

Quizzer is a multi-user web application designed to help users prepare for exams through structured quizzes. The platform allows administrators to manage subjects, chapters, quizzes, and users while providing an engaging quiz experience for users.

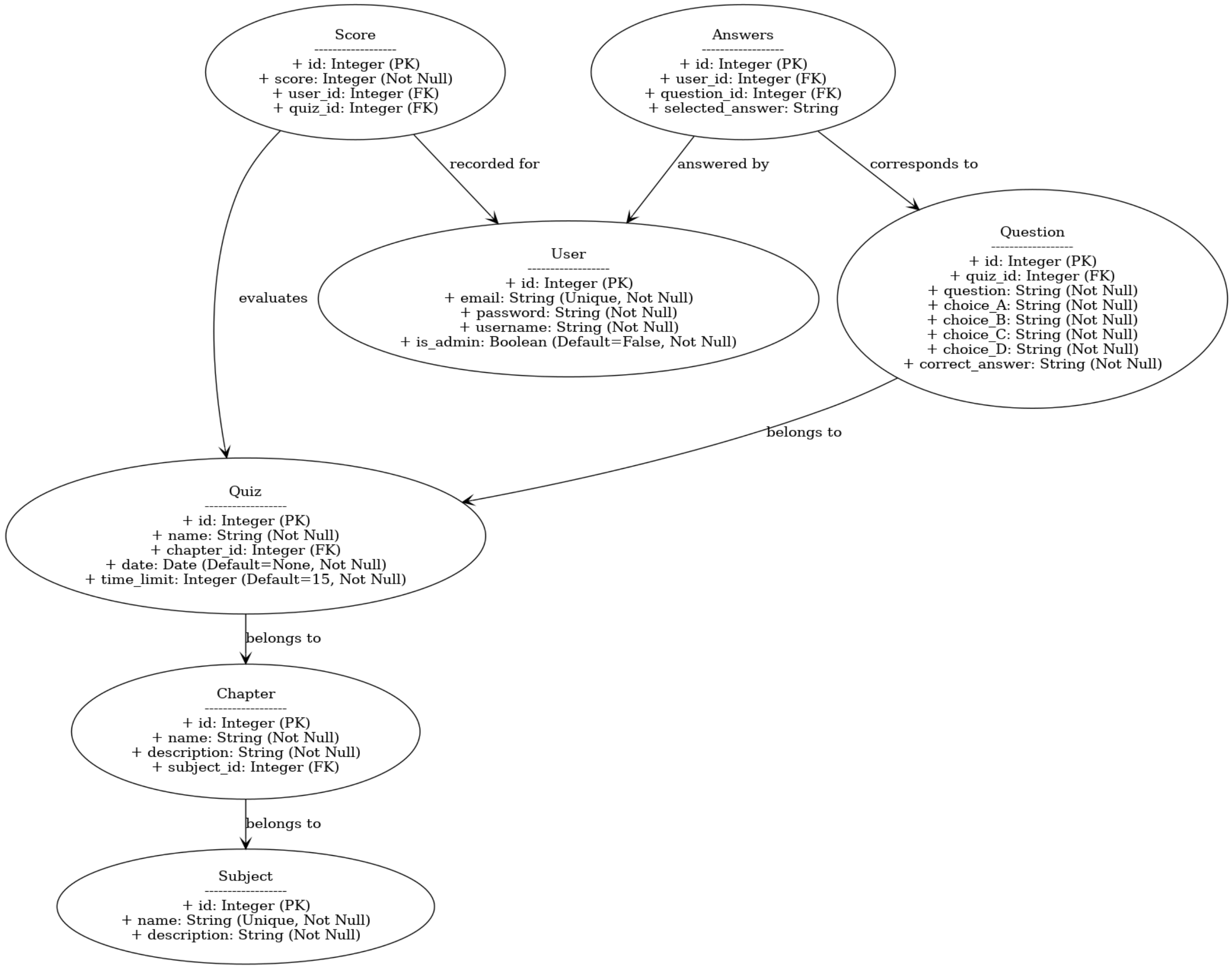
Technology Stack

- **Back-End:** Flask (Python-based web framework)
- **Front-End:** Jinja2 templating, HTML, CSS, Bootstrap, Javascript
- **Database:** SQLite
- **Authentication:** Flask-Login, WTForms
- **ORM:** SQLAlchemy

Data Model and Database Design

- **User:** Stores user details and roles.
- **Subject:** Represents subjects created by the admin.
- **Chapter:** Stores chapters under subjects.
- **Quiz:** Contains quizzes under chapters.
- **Question:** Stores multiple-choice questions.
- **Score:** Keeps records of user quiz attempts.

- **Answers:** Stores user-selected answers.


Score

+ id: Integer (PK)
+ score: Integer (Not Null)
+ user_id: Integer (FK)
+ quiz_id: Integer (FK)
Answers

+ id: Integer (PK)
+ user_id: Integer (FK)
+ question_id: Integer (FK)
+ selected_answer: String
recorded for
answered by
corresponds to
User

+ id: Integer (PK)
+ email: String (Unique, Not Null)
+ password: String (Not Null)
+ username: String (Not Null)
+ is_admin: Boolean (Default=False, Not Null)
Question

+ id: Integer (PK)
+ quiz_id: Integer (FK)
+ question: String (Not Null)
+ choice_A: String (Not Null)
+ choice_B: String (Not Null)
+ choice_C: String (Not Null)
+ choice_D: String (Not Null)
+ correct_answer: String (Not Null)
evaluates
belongs to
Quiz

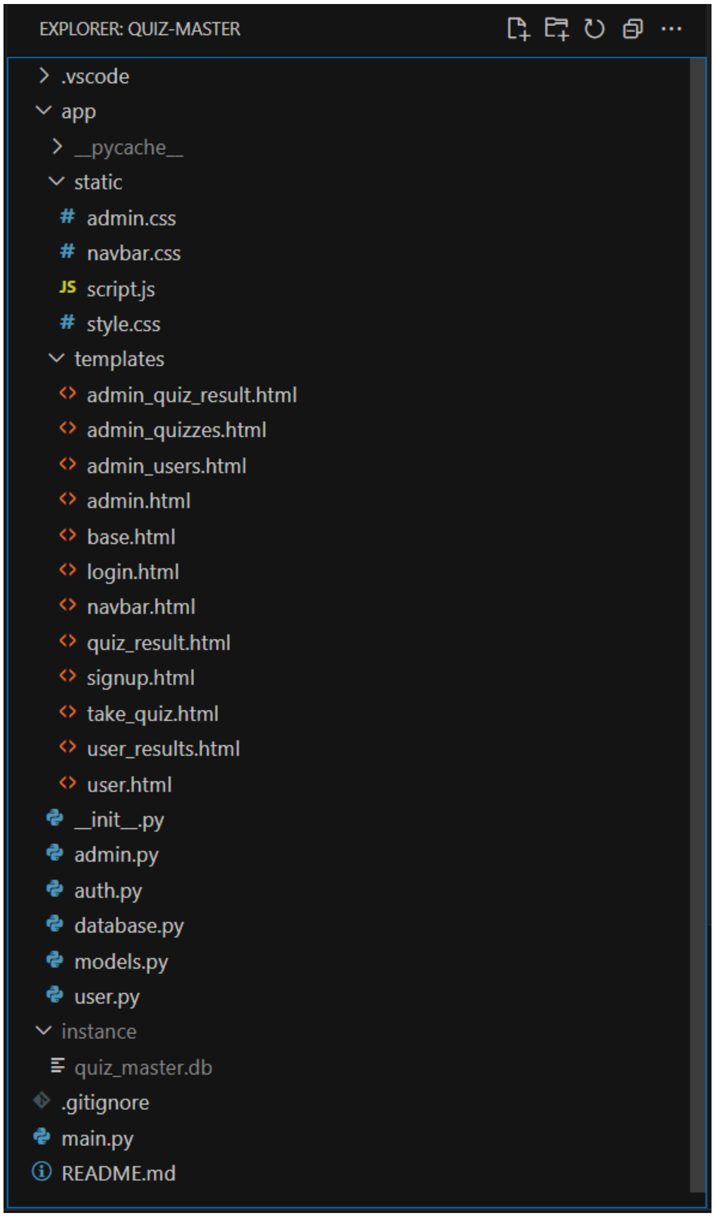
+ id: Integer (PK)
+ name: String (Not Null)
+ chapter_id: Integer (FK)
+ date: Date (Default=None, Not Null)
+ time_limit: Integer (Default=15, Not Null)
belongs to
Chapter

+ id: Integer (PK)
+ name: String (Not Null)
+ description: String (Not Null)
+ subject_id: Integer (FK)
belongs to
Subject

+ id: Integer (PK)
+ name: String (Unique, Not Null)
+ description: String (Not Null)

All relationships follow a structured hierarchy, ensuring referential integrity and cascading deletes.

Folder Structure:



EXPLORER: QUIZ-MASTER

```
> .vscode
v app
  > __pycache__
  v static
    # admin.css
    # navbar.css
    JS script.js
    # style.css
  v templates
    <> admin_quiz_result.html
    <> admin_quizzes.html
    <> admin_users.html
    <> admin.html
    <> base.html
    <> login.html
    <> navbar.html
    <> quiz_result.html
    <> signup.html
    <> take_quiz.html
    <> user_results.html
    <> user.html
  __init__.py
  admin.py
  auth.py
  database.py
  models.py
  user.py
v instance
  quiz_master.db
.gitignore
main.py
README.md
```

Features

Administrator (Quiz Master)

- Has root access to manage users and content.
- Can create, update, and delete:
  - Subjects
  - Chapters under subjects
  - Quizzes under chapters

    - Questions under quizzes
- View quiz results
- Set date and duration of the quiz

User

- Registers and logs into the platform.
- Attempts quizzes and submits answers.
- View scores and quiz history.

Core Features

- User Authentication: Secure login and registration.
- Admin Dashboard: Interface for managing users, subjects, quizzes, and questions.
- Quiz Attempt System: Timed quizzes with multiple-choice questions.
- Score Tracking: User quiz scores stored for historical reference.
- Search : Admin can search for users, quizzes, and subjects while users can search for subjects
- Timer functionality during quiz
- Front-end form validation using JavaScript and HTML5.
- Flask extensions for authentication and security.
- Styled interface using Bootstrap for enhanced UI/UX.
- Flask-Login for session management and role-based access control.

Conclusion

Quizzer provides an effective platform for exam preparation with a structured and user-friendly interface. By leveraging Flask and SQLite, the application ensures efficient management of quizzes and users. Further improvements could include additional authentication security and real-time quiz analytics.

Video Link :

https://drive.google.com/file/d/1SHWWPiFy5WQHMstkLOYTx6HJ0SY8ZdFt/view?usp=drive_link