

# Quiz Master - Project Report

## 1. Student Details

- **Name:** J Angeline Hephzibah
  - **Roll Number:** 23f3004447
- 

## 2. Project Details

**Project Title:** Quiz Master

**Problem Statement:**

The objective of this project is to create an online **quiz management system** where:

- **Teachers** can create subjects, chapters, quizzes, and questions. They can also view student performance through stats.
- **Students** can browse subjects, take quizzes, and view their quiz history.

**Approach:**

- **Database Design:** Structured with **subjects**, **chapters**, **quizzes**, **questions**, and **results**.
  - **Backend:** Flask for API handling.
  - **Frontend:** HTML, CSS, and JavaScript for dynamic pages.
  - **User Roles:** Teacher (admin-like functionality) and Student.
- 

## 3. Frameworks and Libraries Used

## Backend (Flask Framework):

- Flask
- Flask-SQLAlchemy (ORM for database)

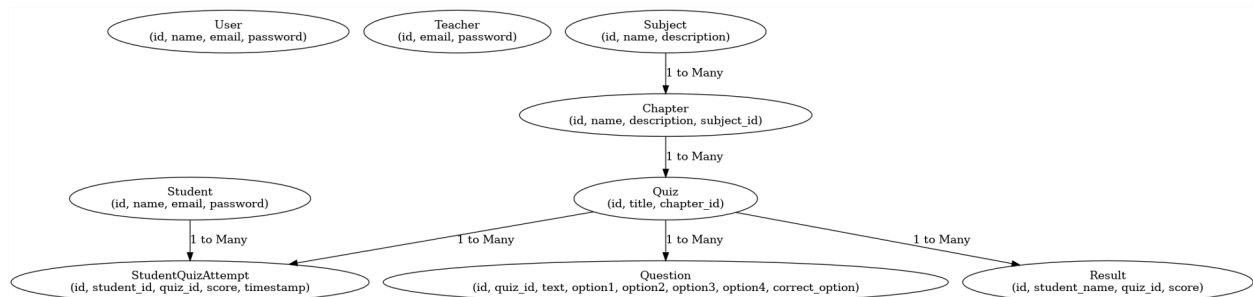
## Frontend:

- HTML
- CSS
- JavaScript

## Database:

- SQLite
- 

## 4. ER Diagram



### Tables Implemented

#### 1. Users (student registration)

- Stores student details (ID, name, email, password).

#### 2. Teachers

- Stores teacher credentials (ID, email, password).

### 3. Subjects (Created by teachers)

- Each subject has an ID, name, and description.

### 4. Chapters (Linked to subjects)

- Each chapter belongs to a subject (foreign key: **subject\_id**).

### 5. Quizzes (Linked to chapters)

- Each quiz belongs to a chapter (foreign key: **chapter\_id**).

### 6. Questions (Linked to quizzes)

- Each question belongs to a quiz and has multiple options and a correct answer.

### 7. Student Quiz Attempts

- Stores student quiz attempts, including their score and timestamp.

### 8. Results

- Stores student quiz scores and links them to quizzes.

- 

---

## 5. API Resource Endpoints

### Student API Endpoints:


- **POST /signup** – Student registration
- **POST /login** – Student login
- **GET /dashboard** – Fetch available subjects
- **GET /chapters/<subject\_id>** – Fetch chapters
- **GET /quizzes/<chapter\_id>** – Fetch quizzes

- `POST /quiz/submit` – Store quiz results

### Teacher API Endpoints:

- `POST /teacher_login` – Teacher login
  - `POST /add_subject` – Create a subject
  - `POST /add_chapter/<subject_id>` – Add a chapter
  - `POST /add_quiz/<chapter_id>` – Add a quiz
  - `POST /add_question/<quiz_id>` – Add quiz questions
  - `GET /quiz_results/<quiz_id>` – View student results
- 

## 6. Presentation Video

 recording.mp4

---

## 7. Conclusion

The **Quiz Master** project successfully provides a platform where teachers can create quizzes, and students can take them while tracking their progress. Future improvements could include **better authentication, real-time leaderboards, and quiz analytics.**