Exam Evaluation Project

Synopsis

Goal of the project is to create an evaluation portal where teachers can upload questons, evaluate and perform other operations as explained in the process section.

Tech Stack

- HTML, CSS, JS
- Python, Django
- Postgres

Design Guidelines

colours, just use base black, whites, grays

▼ Process

1. User Stories and Roles:

1.1 Primary Users:

- **Students (Aged 10 - 18):**
  - *Goals:*
    - Log in to the application.
    - Attempt questions assigned by teachers.
    - Receive a score for each attempt.
    - Have the ability to retry questions multiple times.
- **Teachers:**
  - *Goals:*
    - Log in to the application.

        - Create questions within the system.
        - Assign questions to individual students or groups.

1.2 Authentication and Authorization:

- **User Authentication:**
    - Simple email address and password system.
- **User Roles and Access:**
    - **Students:**
        - Log in and answer questions.
    - **Teachers:**
        - Log in, create and assign questions.
    - **Admins:**
        - Log in, add and manage teacher accounts.

2. Lesson and Task Management:

2.1 Lesson Creation and Management:

- Lessons are containers of questions.
- Lessons are created by teachers.
- Teachers can edit and manage the content of lessons.
- Lessons/quizzes are assigned to specific cohorts or classes.

2.2 Task Assignment:

- Tasks are created and managed by teachers.
- Tasks are assigned to specific lessons.
- Teachers can assign tasks to entire cohorts or individual students within a cohort.

2.3 Task Availability:

- Tasks are available to students at all times.
- Students can access and attempt tasks based on their cohort assignments.

- There is no scheduling of tasks; they are constantly accessible.

3. Question Categories:

- Specify the requirements and functionalities for each question category (1-8).
- Define any specific rules or constraints for each category.
- Matching 4 audio clips to images:
  - Students match four audio clips to corresponding images.
- Matching 4 audio files to 4 audio files:
  - Students match four audio files to four other audio files.
- Pick 1 of 4 audio for 1 image:
  - Students choose one audio clip from a set of four that corresponds to a given image.
- Audio to correct word:
  - Students match audio clips to the correct associated words.
- Eliminate images based on sounds:
  - Students eliminate images based on the sounds they hear.
- Arranging audio segments in order:
  - Students arrange audio segments in the correct order.
- Text fill in the blanks:
  - Students fill in the blanks in a given text.
- Spelling question:
  - Students answer questions related to spelling.

4. Multimedia Handling:

4.1 File Upload:

- Audio clips and images are uploaded through the front end of the site.

4.2 Storage and Referencing:

- Uploaded files are stored as static files.

- References to these files are maintained in the database.

4.3 Supported File Formats:

- **Audio:**
  - Supported formats include MP3 and WAV.
- **Images:**
  - Supported formats include PNG, JPG, and JPEG.

5. Task Completion and Grading:

5.1 Task Completion Criteria:

- A task is considered complete when the student achieves a score of 100%.
- To complete a task, the student must provide the correct answer, which should be uploaded with the question.

5.2 Grading System:

- Tasks are graded using a simple direct comparison method.
- The system compares the student's answer with the correct answer stored in the system.

5.3 Scoring:

- Scoring is binary: either correct (100%) or incorrect (0%).
- No partial credit is awarded

Milestones

This whole project is divided in 2 milestones

Milestone 1: Sections containing Teachers and tasks will be developed here

Milestone 1: Sections containing Students and Scores and task assignment will be developed