CS1660 Final Project Workflow

Website URL : https://cs1660-term-project-663294651398.us-central1.run.app/

Git Repo URL: https://github.com/naruto-sai/CS1660-Term-Project

1. Uploading the CSV File

- Task:
    - The instructor uploads a CSV file containing:
        - Student IDs
        - Course Name
        - Start Date
    - The file's data is processed and stored in the database.
- Trigger:
    - The CSV upload triggers a backend function to parse the file and store the data in Firestore.
- Deliverable:
    - Student IDs, course name, and start date are saved in Firestore under the relevant course/session structure.

2. QR Code Generation

- Task:
    - A scheduled function is triggered 10-15 minutes before the start time.
    - The function:
        - Generates a QR code with a unique session ID for the day.
        - Uploads the QR code to a Cloud Storage bucket.
        - Creates a new session for the course in the database.
        - Copies the student IDs from the course roster to the session instance with their attendance marked as false.
- Trigger:
    - Cloud Scheduler triggers the function based on the course start time.
- Deliverable:
    - A QR code is generated, uploaded, and associated with the session in the database.
    - A new session instance is created with all students’ attendance set to false.

3. Marking Attendance

- Task:
    - Students scan the QR code using a front-end interface.
    - They enter their student ID and submit it.
    - The backend API validates the session ID and student ID and updates the database to mark the student's attendance as true.
- Trigger:
    - Frontend Action: API is triggered when the student submits their ID.
- Deliverable:
    - The student's attendance for the session is updated to true in Firestore.

4. Displaying Attendance

- Task:
    - A function (get_attendance) retrieves and displays the attendance data for the session, showing:
        - All student IDs.
        - Their current attendance status (true or false).
- Trigger:
    - Manual Query: This is triggered on-demand via the instructor's dashboard or an API request.
- Deliverable:
    - A list of all students and their attendance status for the session.

5. Rebuilding on New Commits

- Task:
    - Use Cloud Build to automatically rebuild and redeploy the backend and frontend whenever a new commit is pushed to the GitHub repository.
- Trigger:
    - Cloud Build Trigger: Initiated on each new commit.
- Deliverable:
    - The application is rebuilt and deployed automatically.

Triggers

Trigger 1: Scheduled QR Code Generation

- What Happens:
    - Cloud Scheduler triggers the QR code generation function 10-15 minutes before the course start time.
    - The QR code is generated, and uploaded, and the session is created with attendance data.
- Tools Used:

    - Google Cloud Scheduler
    - Google Cloud Functions
    - Firestore Database

Trigger 2: Rebuild on New Commits

- What Happens:
    - Cloud Build detects a new commit to the GitHub repository.
    - Automatically rebuilds and redeploys the backend and frontend.
- Tools Used:
    - Google Cloud Build
    - GitHub Triggers