1. Project Progress Update

As of 7 May 2025, the Student Task Manager project has made significant progress toward its initial milestones outlined in Task 4.3HD.

The original plan targeted Weeks 1 – 6 for requirement definition, backend development (API and Firestore), and frontend UI creation.

Due to time constraints, development was condensed into 5 days, resulting in a simplified local version of the application.
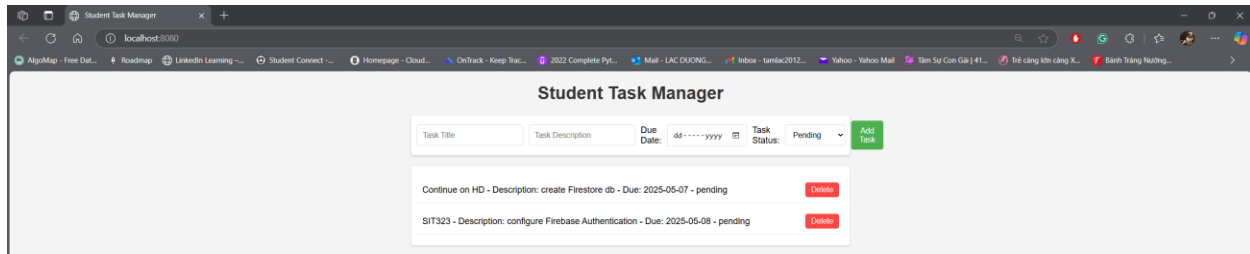
 Completed Tasks

- **Backend Development**: Implemented a RESTful API using Express.js with mock data (an array) instead of Firestore, running on localhost:8080.
    - Supports GET, POST, and DELETE operations for tasks.

- **Frontend Development**: Built a responsive UI using vanilla JavaScript, HTML, and CSS with Flexbox. Features include:
    - Task listing.
    - Adding tasks via a form.
    - Deleting tasks with a confirmation prompt.

- **UI Enhancements**: Added a white background, shadows, and rounded corners to the form and task list for better aesthetics.

Deviations and Challenges

The original plan (4.3HD) intended to use Google Cloud Platform (GCP) services like Firestore for data storage and App Engine for hosting by Week 4. However, due to the compressed timeline (5 days instead of 6 weeks), a local setup with mock data was prioritized to demonstrate core functionality (CRUD operations).

- Challenges included an addEventListener error due to mismatched IDs (resolved by synchronizing IDs to tasks) and UI issues (e.g., adjusting the form's white border to contain the "Add Task" button, which was not fully resolved).

Evidence of Progress

2. Updated Project Plan

The original plan (4.3HD) has been adjusted due to the initial focus on a local setup. The updated timeline leverages the remaining time until the final submission deadline (27 May 2025) to implement the cloud-native architecture as originally intended.

Changes to the Original Plan

- Delayed GCP Integration: Originally planned for Weeks 1–4, GCP integration (Firestore, App Engine) was deferred. A local Express.js server with mock data was used instead.
- Simplified Scope: Authentication and Cloud Functions for reminders were postponed to prioritize core CRUD functionality in the initial phase.

Revised Timeline (7 May– 27 May 2025)

- 7 – 10 May: Set up GCP environment, create Firestore database, and configure Firebase Authentication for Google login.

- 11 – 15 May: Migrate backend to Firestore, associating tasks with user IDs (ownerID).

- 16 – 20 May: Update frontend to use Firestore SDK and integrate authentication for user-specific task management.

- 21 – 25 May: Deploy backend to App Engine, implement Cloud Functions for task reminders (in-app and email).

- 26 – 27 May: Perform testing, optimization, and submit the final cloud-native application