**Project Title: Trainee Mini project Tracker**

**Project Overview**

You are building an **Trainee Mini project Tracker** web application where Trainees can view, update, and track their mini projects, while Trainers can assign, edit, or delete mini projects. The frontend is built using **React / HTML** and Axios / Fetch is used to interact with the backend API for mini project-related operations.

This project mirrors real-world corporate workflows and provides ample scenarios to apply and practice Axios functionalities.

**Detailed Requirements**

**Installing Axios**

* Install Axios using npm install axios.
* Create a reusable Axios instance (axios.js) with global configurations like base URL and headers.

**Making API Requests with Axios**

* Retrieve mini project lists assigned to Trainees.
* Send new mini projects to the backend.
* Update existing mini projects.
* Delete mini projects once completed or obsolete.

**Handling Responses and Errors**

* Show loading indicators during API calls.
* Display success messages when mini projects are added or updated.
* Handle validation errors (e.g., missing fields) and server errors gracefully.
* Notify users when requests fail due to network issues.

**Configuring Axios Requests**

* Include authentication headers like JWT.
* Use query parameters to filter mini projects by status, priority, or due date.

**GET Request**

* Fetch mini projects assigned to an Trainee.
* Example: GET /mini projects?status=inprogress&priority=high.

**POST Request**

* Trainers can assign mini projects by sending mini project data like title, description, assigned Trainee, priority, and due date.

**PUT Request**

* Trainees can update mini project progress or details.
* Example: Mark a mini project as complete or change the deadline.

**DELETE Request**

* Trainers can remove mini projects that are no longer required.
* Include confirmation before deleting.

**Advanced Axios Usage**

* Use Axios interceptors to:
  + Automatically attach authentication tokens.
  + Log requests and responses.
  + Retry requests in case of network errors.
* Implement cancel tokens to abort unnecessary requests when switching between mini project views.

**Global Axios Configuration**

* Configure Axios instance (axios.js) with:
  + Base URL for API endpoints.
  + Default headers like Content-Type: application/json.
  + Request and response interceptors to handle loading states and errors.

**User Roles**

1. **Trainee** can:
   * View assigned mini projects.
   * Update mini project status or details.
   * Filter mini projects by due date or priority.
2. **Trainer** can:
   * Create mini projects and assign them.
   * Edit or delete mini projects.
   * View reports of mini project completion.

**Note: Use React, Tailwind + Django / HTML, CSS, Bootstrap or Tailwind, JS + Django**