IMPORTANT NOTE: Unless mentioned below, all assumptions are open to your interpretation. Please make your own assumptions and build this system. We do not entertain any queries or clarifications associated with the problem statement below. As a team, you are open to drawing your own boundaries. Your grade has no impact on incompleteness of your assumptions.

Problem Code: ResearchNest

Problem Statement: You are tasked with developing a prototype of the ResearchNest portal, focusing on the Student Dashboard and a simplified Faculty/Admin Dashboard. The goal is to deliver a functional 48-hour prototype that implements core features of the Progress Tracker and Student Profile Page for both user types.

Skills: Your team will be assessed on Git proficiency, database design (SQL & NoSQL), full-stack development with MERN, and problem-solving in handling complex data relationships and workflows.

Feature(s): Create a Student Task Progress Tracker. It is a hierarchical system that organizes a student's academic journey into milestones, stages, tasks, and subtasks, each with statuses of "Locked," "In Progress," or "Completed." A parent item's completion depends on its children, meaning milestones require all stages completed, stages require all tasks completed, and tasks require all subtasks completed.

Example: For example, the **"Course Completion"** milestone may include stages like *Research Methods* or *Advanced Databases*, each containing tasks such as *Research Paper Submission*, *Tutorial Presentation*, or *Assignment 1*, which can be further broken into subtasks like *Draft Abstract*, *Prepare Slides*, or *Assignment 1 – Part A*. A milestone like **"Thesis Submission"** might include stages such as *Proposal Approval* and *Final Defense*, with tasks like *Write Literature Review* and subtasks like *Collect References* or *Summarize Related Work*.

Deliverables:

Part 1: Data Modeling & Querying (30%)

Design SQL or NoSQL schemas for the Progress Tracker and Student Profile, ensuring they capture the hierarchy of milestones, stages, tasks, and subtasks with appropriate status fields. Write the required SQL or MongoDB queries for inserting, retrieving, and updating records, and then compare both approaches to recommend the most suitable database for ResearchNest.

Part 2: MERN Stack Implementation (60%)

Build a MERN stack application with login, a Student Dashboard, and a simplified Faculty/Admin Dashboard that connects to the database chosen in Part 1. Implement a REST API for authentication, data retrieval, and updates, while ensuring role-based views where students manage their own progress and faculty can oversee and override student milestones.

Part 3: Git & Collaboration (10%)

Demonstrate effective use of Git by setting up a shared repository, following a branching strategy, and maintaining a clear commit history. The team must also intentionally create and resolve a merger conflict, documenting the resolution process as proof of collaborative development.

Submission Criteria:

- **Source Code:** Git repo link with all frontend and backend code (commit history reviewed).
- **sql_and_nosql_solutions.md** with schemas, queries, and analysis. Include this as part of your Git Repo.
- **README.md** with MERN implementation details and set-up steps along with short details on design decisions, solution diagram.
- **Submit Peer Review Form**: https://forms.office.com/Pages/ResponsePage.aspx?id=vDsaA3zPK06W7IZ1VVQKHFzW4INMf2JMjyL9qPnlPbNURjRUSDA0QURKWFBCTzVCNFdDU1pDVTJZSy4u
- **LLM Usage Form**: https://forms.office.com/Pages/ResponsePage.aspx?id=vDsaA3zPK06W7IZ1VVQKHFzW4INMf2JMjyL9qPnlPbNUNFFSOTM3MDE4T0NPWUpNOTZKSFFQSUJTRC4u