FULL STACK PROBLEM STATEMENTS

Problem Statements (PS)

PS-01: Build a collaborative project management system.
Features:
   a) User Management should include: students dashboard and teacher dashboard.
   b) Team registration (One login for all team members)
   c) Team dashboard should include project title registration, and relevant documents upload.
   d) Status page to check project status (Page can be like a blog with request responses in it).
   e) Teacher dashboard should consist for listed registered projects.
   f) Blog like interface on selecting a particular registered project with an option to communicate with the team and add status to be visible to student batch.
   g) Evaluation module to grade projects.
   h) Statistics module of with number of projects submitted, approved, can be included.

PS-02: Build a web application such as a teacher – student chat bot.
Features:
   i) User Management should include: students dashboard and teacher dashboard.
   j) User registration (Option to select student / teacher)
   k) Student dashboard can have option to select any teacher and pose a question and gets the response once its been answered (should have file upload feature).
   l) Teacher dashboard can have a consolidated questions posted by the students which can be answered by the teacher with either a texting or by uploading a file.
   m) Statistics module of number of active students, teachers can be included.

Guidelines
A problem statement will be provided in the challenge. You have to build a functional app. Based on the question, it should be a web app
Each problem comprises the following:
   a) Essential feature sets that you need to build
   b) List of good-to-have features
   c) Visual description of the problem statement
Note: While these are not mandatory, you can earn extra points if you build them.

After you have built your application, you must do the following:
   a) Zip the source files
   b) Upload the zipped file via the upload option that is specified in the problem statement.

The submissions will be evaluated manually by an expert based on the following parameters:
   a) Functionality of the app
   b) Design aesthetics
   c) Code quality

Essential features
   a) Design should be desktop and portable device friendly.
   b) Form validation at user and server level.
   c) Notification at appropriate actions.
   d) Integrate User Management
   e) Object Oriented Programming Approach for server scripts.
   f) Structured file system with restricted user access.
   g) No restrictions over Programming language i.e. you can use any scripting language.