Project Management App

A company is managing its projects using a mobile application. Project managers record project details, and team members can view and interact with the project information.
On the server side, at least the following details are maintained:

- Id - the internal project id. Integer value greater than zero.
- Name - A string of characters representing the project name.
- Team - A string of characters representing the project team name.
- Details - A string of characters having the project details.
- Status - A string of characters representing the project status. Eg. “planning”, “in progress”, “on hold”, “completed”, etc.
- Members - An integer value representing the number of team members involved in the project.
- Type - the project type. A string of characters. Eg. “software development”, “marketing”, etc.

The application should provide the following features (available without restarting the app):

● Project Manager Section (separate activity/screen)

A. (1p) Create a new project. Using **POST /project** call by specifying all the project details. Available both online and offline.

B. (2p) View all the projects in the system, in a list. Using **GET /projects** call, the project manager will retrieve all of them. The list should display at least the id, name, team, and type. If offline, the app will display an offline message and a way to retry the connection and the call. Once retrieved, the data should be available on the device, regardless of whether online, offline, or restarts.

C. (1p) By selecting a project from the list, the project manager will be able to view all the project details. To retrieve all the project details, **GET /project** call will be used by specifying the project id. Once retrieved, the data should be available on the device, regardless of whether online, offline, or restarts.

● Team Member Section (separate activity/screen) - Available online only.

A. (1p) View all the projects that are in progress in the system in a list. Using **GET /inProgress** call, the team member will retrieve all the projects having this status.

B. (1p) By selecting a project from the list, the team member will be able to enroll in the ongoing work. Using **PUT /enroll** by specifying the project id the user will be able to enroll in the project.

● Analytics Section (separate activity/screen) - Available online only.

(1p) Using **GET /allProjects** call, create a top 5 projects by the number of members. The list will present the projects in ascending order by status and descending by the number of team members. If no such projects are available, the application will display a proper message. Also, note that the server is not ordering the list in any way.

● (1p) On the server side, once a new project is added to the system, the server will send, using a WebSocket channel, a message to all the connected clients/applications with the new project object. Each application that is connected will display the received project details, in a human form (not JSON text or toString) using an in-app “notification” (like a snack bar or toast or a dialog on the screen).

(0.5p) On all server or DB operations, a progress indicator will be displayed.

(0.5p) On all server or DB interactions, if an error message is received, the app should display the error message using a toast or snackbar. A log message should be recorded on all interactions (server or DB calls).