

Project App

A team is managing their project ideas using a mobile app. The users are able to view all the ideas and promote them to projects.

On the server side at least the following details are maintained:

- Id - the internal project id. Integer value greater than zero.
- Name - the project name. A string of characters representing the project name.
- Budget - the initial estimated budget. An integer value greater than zero.
- Type - the project type. Eg. "small", "medium", "large". A string type.
- Status - the project status. Eg. "idea", "pending", "approved", "discarded". A string type.

The application should provide at least the following features:

- Idea Section (separate activity - available offline too)
 - a. (1p) View the available ideas. Using **GET /ideas** call, the user will receive the list of ideas available in the system. If offline the app will display an offline message and a way to retry the connection and the call. For each idea the name, budget and the type are displayed.
 - b. (0.5p) Add a new idea. The user will be able to add a new idea, using a **POST /add** call, by specifying the idea name type and budget. Available only when online. The server will return the idea object with a populated id.
 - c. (0.5p) Remove idea. While online, the client will be able to purge an idea using **DELETE /delete** call by specifying the idea id. The server will return the idea object.
 - d. (0.5p) Both offline and online the client will be able to view the list of persisted ideas.
 - e. (0.5p) Remove local persisted ideas.
 - Project Section (separate activity - available only online)
 - a. (1p) The list of all the projects. The list will be retrieved using the **GET /projects** call, in this list along with the name, budget and type, the app will display the status also.
 - b. (1p) Promote an idea to a project. Retrieved the list of ideas using the **GET /ideas** call. Using **POST /promote** call by specifying the idea id, the user will be able to convert an idea to a project. The status of the new project will be pending. The server will return the project.
 - c. (0.5p) Approve a project. Using a **POST /approve** call, by sending the project id, the user will approve the project. The project will have the approve status. On success 200 OK status will be returned along with the project.
 - d. (0.5p) Discard a project. Using a **POST /discard** call, by sending the project id, the user will discard the project. The project will have the discard status. On success 200 OK status will be returned along with the project.
 - e. (0.5p) Delete a project. Using **DELETE /remove** call, by sending a valid project id, the server will remove the project. On success 200 OK status will be returned along with the project.
- (0.5p) On all server operations a progress indicator will be displayed.
- (0.5p) On all server interactions, if an error message is received, the app should display the error message using a toast or snackbar.
- (0.5p) On all interactions (server or db calls), a log message should be recorded.