

Task Management App

A group of users is tracking their daily tasks and productivity using a mobile application. Each user can manage their task data.

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

- Id - the internal task data-id. Integer value greater than zero.
- Date - the date when the task was recorded. A string in the format "YYYY-MM-DD".
- Type - the type of task performed. A string of characters.
- Duration - the duration spent on the task. A decimal value.
- Priority - the priority level of the task. A string of characters.
- Category - the category of the task. A string of characters.
- Description - a description of the task. A string of characters.

The application should provide at least the following features:

- Main Section (separate activity)

A. (1p) View the list of recorded tasks days. Using the **GET /taskDays** call, the user will retrieve the list of all their recorded tasks. 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 or offline.

B. (2p) By selecting a date, the user can view the details of the tasks recorded on that date. To retrieve the details of a specific date's tasks, the **GET /details** call can be used by specifying the date. Once retrieved, the data should be available, regardless of whether online or offline.

C. (1p) Add a new task entry. Using **POST /task** call by specifying all the task details, the user will be able to create a new task record. Available online only.

D. (1p) Delete a task entry. By selecting a task from the list, and using the **DELETE /task** call, the user will be able to delete a task record. Available online only.

- Progress Section (separate activity)

(1p) View the total duration for each month. The list will be retrieved using the **GET /entries** call. The list should display the month and the total duration per month in descending order.

- Top Section (separate activity)

(1p) View the top 3 categories. Using the same **GET /entries** call, compute the top 3 categories by the number of tasks. The list should contain the category name and the number of tasks in descending order.

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

- (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. A log message should be recorded on all interactions (server or DB calls).