# Gym Tracker App

A group of users is tracking their gym activities using a mobile application. Each user is able to manage their own gym activities.

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

- Id - the internal activity id. Integer value greater than zero.
- Activity Name - the name of the gym activity. A string of characters representing the activity name.
- Description - a brief description of the activity. A string of characters.
- Category - the activity category. A string of characters. Eg. “weightlifting”, “cardio”, “yoga”, etc.
- Date - the date the activity was completed. A date value.
- Time - the time spent on the activity. An integer value represents time in minutes.
- Intensity - the intensity level of the activity. A string of characters. Eg. “easy”, “medium”, or “hard”.

The application should provide at least the following features:

- Main Section (separate activity)
  - A. (1p) View the categories available in the system in a list. Using the **GET /categories** call, the user will retrieve the list of all gym activity categories found in the system. If offline, the app will display an offline message and a way to retry the connection and the call. Once retrieved it should be available offline.
  - B. (2p) By selecting a category, the user will be able to get to the list of activities that are having that category. To retrieve the list of activities having the specified category the **GET /activities** call can be used by specifying the category. Once retrieved the list should be available offline.
  - C. (1p) Add a gym activity. Using **POST /activity** call by specifying all the activity details the user will be able to create a new activity. Available online only.
  - D. (1p) Delete a gym activity. By selecting an activity from the list, and using the **DELETE /activity** call, the user can delete a gym activity. Available online only.
- Intensity Section (separate activity)
  - A. (1p)The list of the top 10 easiest activities sorted ascending by intensity level and category. The list will be retrieved using the **GET /easiest** call, in this list along with the name, description, date, and time, the app will display the current intensity level and the category. Note that from the server you are retrieving all the activities.
  - B. (1p) Change the intensity level of gym activity. From the above list, the user should be able to select an activity and change its intensity level using the **POST /intensity** call by specifying the activity id and the new intensity level.
- (1p) On the server side once a new gym activity is added to the system, the server will send, using a WebSocket channel, a message to all the connected clients/applications with the new activity object. Each application, that is connected, will display the received activity 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. On all interactions (server or DB calls), a log message should be recorded.