Pet Owners App

A group of pet owners uses a mobile application to connect and share information about their pets. Each user is able to manage their own pet profile.

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

- Id - the internal pet profile id. Integer value greater than zero.
- Name - the name of the pet. A string of characters.
- Breed - the breed of the pet. A string of characters.
- Age - the age of the pet in years. Integer value.
- Weight - the weight of the pet in pounds. Integer value.
- Owner - the name of the pet owner. A string of characters.
- Location - the location of the pet owner. A string of characters.
- Description - any additional information added by the owner about the pet. A string of characters.

The application should provide at least the following features:

● Main Section (separate activity)

   A. (1p) View the list of pet profiles. Using the **GET /pets** call, the user will retrieve the list of all pet profiles and display the name. 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, even offline.

   B. (2p) By selecting a pet profile, the user can view the details of the pet. To retrieve the details of a specific pet's profile, the **GET /pet** call can be used by specifying the pet id. Once retrieved, the data should be available, even offline.

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

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

● Search Section (separate activity)

   (2p) View pets based on certain search criteria, such as breed, age, and location. The list will be retrieved using the **GET /search** call. The list should present the pets descending by weight, and ascending by age.

● (1p) On the server side, once a new pet profile is added to the system, the server will send, using a WebSocket channel, a message to all the connected clients/applications with the updated pet profile object. Each application, that is connected, will display the received pet profile 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).