

## Hotel

A chain of hotels are sharing their rooms online using a mobile app. For each hotel a manager will be able to add, delete and change the room details. The clients will be able to view all the available rooms, choose one to book and once a room is booked to release it.

On the server side at least the following room details will be maintained:

- Id - the internal room id. Integer value greater than zero.
- Name - the room name. A string of characters representing the room name.
- Type - the room type. Eg. "double", "single", "conference", "apartment".
- Status - the room status. Eg. "available", "taken".
- Size - expressed in square meters. Also an integer value. Eg. 25, 30, etc.

The application should provide at least the following features:

- (3p) Manager Section (separate activity - available only online)
  - a. (1p)(**2p**) The list of available rooms, at least the name, type and room status should be displayed. The list will be retrieved from the server side using a GET /rooms call.
  - b. (0.5p)(**1p**) Add a room. Using a POST /add call, by sending the room details a room will be added in the list, on success the server will return the room object with the id field set. The name + type fields are identifying a room.
  - c. (0.5p) Delete a room. Using POST /remove call, by sending a valid room id, the server will remove the room. On success 200 OK status will be returned.
  - d. (1p) Update the room details. Using POST /update call, by sending a valid room object, the server will update the room represented by the room id.
- (3.5p) Client Section (separate activity - available offline too)
  - a. (1p)(**2p**) View the available rooms. Using GET /available call. If offline the app will display a message with the hotel contact phone number and a way to retry the call.
  - b. (0.5p)(**1p**) Book a room. The client will be able to reserve a room, if available, using a POST /book call by specifying the room id. The action is available only while online.
  - c. (**1p**) Once the client booked a room, instead of presenting the list of the available rooms the app will display the details of his booked room, even when offline. Once online the user will be able to checkout. By doing a POST /checkout with a valid room id the room will be released. At this point the app will display the available rooms again.
  - d. (0.5p) View the local history with all his rooms. For each room that was once booked the app will display the room name and the time when the room was booked.
  - e. (0.5p) Remove the local history.

(**1p**) On the server side once a new room is added in the system the server will send, using a websocket channel, a message to all the connected applications with the new room object. The application will add the new object in the list of available rooms.

(**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, a log message should be recorded.

Only the requirements marked with the **bold** are mandatory, if your laboratory mark is greater or equal to 5.