

Access App

A private corporation is managing its employee access using a mobile application. The security staff is maintaining the access details and the employees are able to view and use their credentials to gain access.

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

- Id - the internal rule id. Integer value greater than zero.
- Name - A string of characters representing the access rule name.
- Level - Integer value greater than zero, representing the security level.
- Status - A string of characters representing the status. Eg. "prepared", "in-use", "canceled", "old", etc.
- From - An integer value representing the beginning (eg. 1610985064), when the rule is in the specified status.
- To - An integer value representing the end, when the rule is in the specified status.

The application should provide the following features (available without restarting the app):

- Staff Section (separate activity)
 - a. (1p) Register a new rule. Using **POST /rule** call by specifying all the rule details. Available online and offline.
 - b. (2p) View all the rules found in the system, in a list. Using **GET /rules** call, the staff will retrieve all of them. The list should display at least the id, name, and level. If offline, the app will display an offline message and a way to retry the connection and the call. Once the list is retrieved it should be available offline and online.
 - c. (1p) By selecting a rule from the list, the staff will be able to view all the rule details. To retrieve the details **GET /rule** call will be used by specifying the rule id. Available online only.
 - d. (1p) Update the rule details. On the details screen, the user will have the option to update all the existing rule fields. Using **POST /update** call with the new rule details. Available online only.
- Employee Section (separate activity) - Available online only.
 - a. (1p) View all the rules in a specified time range (from-to). Using the same **GET /rules** call, the app will retrieve all the rules and will present only the ones included in the specified range.
 - b. (1p) View all the rules at a specified level. Using **GET /level**, by specifying the desired level, will display all the received rules. The list should be ordered by from, to, and status fields all ascending.

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

(0.5p) On all server/DB operations a progress indicator will be displayed.

(0.5p) On all server/DB interactions, if an error message is generated, the app should display the error message using a toast or snack bar. On all interactions (server or DB calls), a log message should be recorded.