

Rezervari

Un policlinica foloseste o aplicatie mobila pentru a gestiona rezervarile pacientilor. Secretariatul confirma rezervarile iar pacientii pot inregistra o noua rezervare.

Pe server se vor memora urmatoarele detalii:

- Id - Identificatorul rezervarii. O valoare intreaga mai mare decat zero.
- Nume - Numele pacientului. O valoare de tip string.
- Doctor - Un string reprezentand numele doctorului.
- Data - Un intreg care reprezinta ziua rezervarii.
- Ora - Un intreg care reprezinta ora rezervarii.
- Detalii - Un string reprezentand detaliile specificate de pacient.
- Status - O valoare booleana care specifica daca rezervarea este confirmata sau nu.

Aplicatia ofera urmatoarele functionalitati (fara a fi nevoie sa restartam):

- Sectiune secretar (ecran separat).
 - a. (2p) Vizualizare rezervari, intr-o lista. Folosind apelul **GET /open**, personalul va accesa toate rezervarile din sistem neconfirmate. Elementele din lista trebuie sa prezinte cel putin: id-ul, numele, doctorul, data, si ora unei rezervari. Ordonate crescator dupa doctor, data si ora. Daca aplicatia este offline, si nu avem rezervari in baza de date locala, se va afisa un mesaj corespunzator si un buton care sa reincerce apelul. Odata ce primul apel reuseste se vor folosi doar rezervarile din baza de date offline/locala.
 - b. (2p) Confirma rezervare. Selectand o rezervare din lista precedenta, folosind apelul **POST /confirm** specificand doar id-ul unei rezervari, se va cere confirmarea unei rezervari. Disponibila online si offline.
 - Sectiune pacient (ecran separat) - disponibil doar online.
 - a. (2p) Inregistrare rezervare. Folosind apelul **POST /create**, specificand toate attributele unei rezervari. Ca raspuns de la server se primeste un intreg reprezentant id-ul rezervarii.
- (1p) Pe server, odata ce se adauga o noua inregistrare, folosind o conexiune de tip WebSocket, se va trimite un mesaj tuturor aplicatiilor conectate. Aplicatiile conectate, vor afisa o notificare cu continutul mesajului primit (folosind un dialog, sau snackbar sau toast). Intr-un format uman, nu text-ul JSON sau toString.
- (1p) Pentru fiecare apel catre server/DB se va afisa un indicator de progres.
- (0.5p) Pentru fiecare interactiune server/DB, daca se produce o eroare, aplicatia va afisa mesajul folosind un toast, snackbar sau dialog.
- (0.5p) Indiferent de apel (server sau DB), se va inregistra un mesaj de debug/log.