

La o clinică privată din Cluj-Napoca încă se fac programările într-un registru. Se cere proiectarea unei aplicații în care să se poată face o programările la toate secțiile din clinică.

1. 1p. La deschiderea aplicației se afișează o fereastră cu toate secțiile (unice) din clinică. O secție este definită astfel:

```
Secție(  
    id,  
    nume,  
    idSefDeSectie (referă spre Medic),  
    pretPerConsultație,  
    duratăMaximăConsultație (ore)  
)
```

O clinică are cel puțin 3 secții (Pediatrie, Psihiatrie, Urologie, Ginecologie etc.) - informațiile despre secții se iau dintr-o bază de date.

2. 2p. Toți medicii din aplicație primesc câte o fereastră proprie, în care pot vedea consultațiile programate.

Un medic poate activa doar într-o secție (cu excepția cazului în care este șef de secție la alta decât cea în care activează)

- să existe cel puțin 10 medici în clinică dintre care cel puțin 3 să fie rezidenți
- informațiile despre medici se iau dintr-o bază de date.

```
Medic(  
    id,  
    idSecție,  
    nume,  
    vechime,  
    rezident(DA/NU)  
)
```

3. 2p. La selectarea unei secții din prima fereastră, se deschide o nouă fereastră în care se poate programa o consultație. Dacă sunt mai mulți doctori alocați pe secție, asistenta poate alege care doctor se va ocupa de pacient.

Programarea unei consultații apare instant pe fereastra medicului corespunzător. (precum și a șefului de secție - opțional)

Data și ora consultației se aleg folosind un DatePicker și (eventual) spinnere.

O consultație este definită astfel:

```
Consultație(  
    id,  
    idMedic,  
    CNPPacient,  
    NumePacient,  
    Data,  
    Ora  
)
```

4. 2p. Se cere afișarea în ordine crescătoare a consultațiilor (pentru fiecare doctor) în funcție de data și ora la care sunt programate. O programare dispare din listă atunci când ora a fost depășită.

Cerinte non-functionale - 2p:

- Validarea datelor de intrare

Exemplu: Nu se poate programa o consultație pe o secție dacă există deja o consultație pe secție care nu se va fi terminat până la ora de început a noii consultații.

- Procesarea va avea loc numai la nivel de service sau de controller; interacțiunea cu sursa de date se va face numai prin intermediul repository-ului (puteti folosi baze de date sau fișiere text)
- Interacțiunea cu utilizatorul va avea loc numai în UI (GUI)
- Se cere eliminarea codului care nu este folosit, precum și a funcționalităților care nu s-au cerut (dacă ați lucrat cu ceva template de la lab)
- Clasele, atributele și metodele lor vor avea exact numele cerute în problema sau nume sugestiv dacă nu s-a specificat explicit numele lor

IMPORTANT

- Se punctează doar cerințele funcționale care rulează
- Orice cod care nu poate fi explicat, atrage după sine nepunctarea cerinței/cerințelor din care face parte
- Nu aveți voie să comunicați în timpul examenului, în nicio modalitate posibilă de a face acest lucru (chat, mail, ...etc.)

Cerințe extra – pt. bonus: 1p în plus la nota finală:

- Orice șef de secție poate vedea într-un tabel separat programările de la secția la care este șef. Dacă un medic nu este șef la nicio secție, nu se va afișa niciun tabel; în schimb, se va afișa numele șefului de secție.
- Asistenta poate șterge o programare. Modificările devin vizibile în ferestrele medicilor.