

## Subiect 18 iunie 2021.

Proiectați și implementați o aplicație client-server pentru următoarea problemă.

Un joc cu 3 jucători numit *Roșu*. Trei utilizatori autentificați pot juca acest joc. La începutul jocului, sistemul va trimite fiecărui participant 3 cărți de joc distincte, alese aleator din multimea {"red.8", "red.9", "red.10", "black.8", "black.5", "black.2", "black.7", "red.7", "red.2"}. Jucătorul care după 3 runde are cele mai puține carti de joc (conform cerinței 2), câștigă jocul. Fiecare utilizator poate să facă următoarele:

1. *Login*. După autentificarea cu succes se deschide o nouă fereastră în care este afișat un buton "Start joc". Doar după ce trei jucători se autentifică în aplicație și apasă butonul de "Start joc", jocul va începe. La începerea jocului, serverul va trimite tuturor jucătorilor id-urile(username) celorlalți jucători și cartile de joc alese pentru jucatorul respectiv. De exemplu: jucatorul 'vasile' primește la pornirea jocului următoarele informații: 'ana', 'ion', {"red.8", "black.7", "red.10"}.
2. *Alege carte*. Fiecare jucător alege o carte și o trimite la server. După ce toți jucătorii au trimis o carte la server, acesta va decide jucatorul (username) care va obține cartile respective, astfel:
  - dacă există o singură carte de culoare roșie, jucatorul care a trimis cartea respectivă va primi toate cartile;
  - dacă există două sau mai multe carti roșii, jucatorul care a trimis cartea roșie având valoarea cea mai mare va primi toate cartile;
  - dacă nu există nicio carte de culoare roșie, jucatorul care a trimis cartea cu valoarea cea mai mică va primi toate cartile.

După verificarea cartilor, serverul trimite tuturor jucătorilor toate cartile primite și numele (username-ul) jucatorului care le-a primit. Aceste informații vor apărea automat pe interfața grafică a fiecărui jucator.

Acest pas se repeta de încă 2 ori. La finalul celor 3 runde, serverul va trimite tuturor jucătorilor numărul de carti obținute de fiecare jucător și clasamentul final. Toți jucătorii vor vedea clasamentul pe interfața grafică ( în ordine crescătoare a numărului de carti obtinute).

3. Un serviciu REST care permite vizualizarea cartilor primite la pornirea jocului de fiecare jucător pentru un anumit joc.
4. Un serviciu REST care permite vizualizarea cartilor trimise de un anumit jucător la un anumit joc și numărul de carti obtinute de acesta la fiecare runda.

Observație:



Cartile primite de fiecare jucator la inceperea jocului, cartea trimisa de un jucator la server la fiecare rundă și numărul de carti obtinute la fiecare runda se păstrează în baza de date.

### Cerințe:

- Aplicația poate fi dezvoltată în orice limbaj de programare (Java, C#, etc).
- Datele vor fi preluate/salvate dintr-o bază de date relațională.
- Pentru o entitate (exceptând jucator) se va folosi un instrument ORM pentru stocarea datelor ( nu se accepta SpringJPA).
- Pentru testarea serviciilor REST se va folosi o extensie a unui browser web/aplicație (Postman, REST client, etc).

**Barem:**

- Login - 0.5p
- Logout -0.5
- Start joc - 0.5p
- Trimiterea unei carti la server (cu adaugare in baza de date) - 0.5 p
- Determinarea jucatorului care va primit cartile - 1p
- Determinarea clasamentului la finalul jocului -1 p
- Actualizarea automata a ferestrelor (pentru toate cazurile) -1.5p
- Serviciu REST 1 - 1p
- Serviciu REST 2 - 1p
- Folosire ORM (cu executie) -1p
- Structura bazei de date si popularea ei -0.5p
- Arhitectura si proiectare -1 p