

# ChessS

Zaharie Robert Gabriel

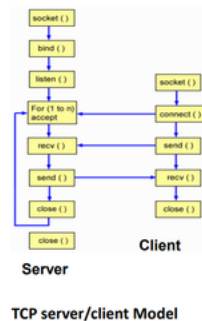
December 6, 2021

## 1 Introducere

Scopul acestui proiect este de a crea o aplicatie server care pune la dispozitia clientilor conectati prin retea o serie de nivele de puzzle-uri tip sah. Aplicatia server va inregistra fiecare miscare propusa de client si o va trece prin etapele predeterminate pt a ajunge la solutia fiecarui puzzle – fiecare nivel va avea un scor ce va fi asignat clientului dupa terminarea nivelului. Evidenta va fi tinuta pe un “leaderboard” actualizat pe baza username-ului(daca exista, scorul acestuia va fi actualizat, iar daca nu exista, username-ul va fi adaugat in leaderboard).

## 2 Tehnologii utilizate

Pentru realizarea proiectului am folosit Transmission Control Protocol (TCP), un protocol de transport orientat, cu conexiune, fara pierdere de informatii, ce controleaza fluxul de date. Pentru a asigura posibilitatea mai multor jocuri petrecute simultan am folosit varianta TCP concurent in defavoarea variantei TCP iterativ, astfel, daca un client este la nivelul 1 acesta nu ar trebui sa astepte alt client care ar fi la un nivel mai avansat.



## 3 Detalii implementare

Functia getUsername() va prelua username-ul introdus de client si-l va trimite catre sever prin sendUsername(), acesta fiind verificat, prin functia LeaderboardSearch() daca exista deja in baza de date. In caz afirmativ, functia LeaderboardSearch() va afisa username-ul si scorul acestuia, iar in caz negativ, va adauga in baza de date noul userID si va afisa scorul 0.

Functia getNewLevel() genereaza o matrice de 8x8 reprezentand nivelul actual preluat prin functia getNextLevel() din baza de date.

Functia playMove() inregistreaza modificarea matricei facuta de client

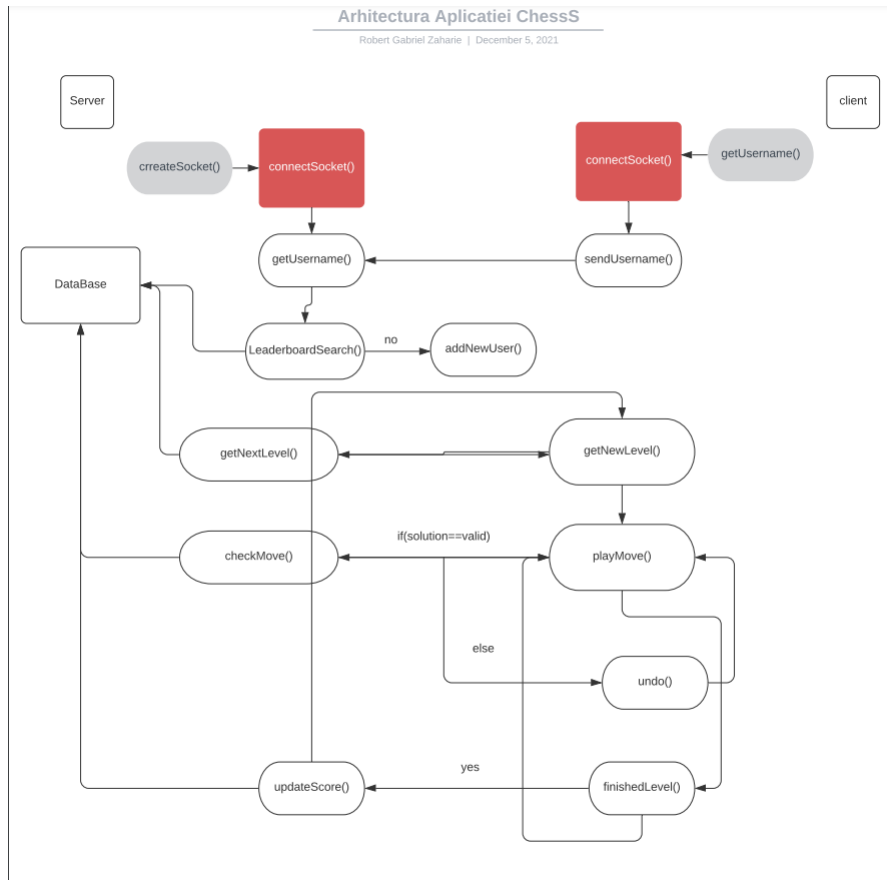
Functia checkMove() verifica modificarea facuta de client, astfel incat matricea obtinuta sa coincida cu urmatoarea matrice a solutiilor din baza de date pt nivelul curent.

Functia undo() modifica matricea tablei de joc la valoarea initiala generata prin getNewLevel(). Functia finishedLevel() verifica daca matricea obtinuta dupa miscarea jucatorului coincide cu ultima matrice a solutiilor corespunzatoare nivelului curent.

Funcția `updateScore()` actualizează scorul clientului în baza de date, în cazul în care nivelul a fost terminat.

Baza de date conține username-urile și scorurile jucătorilor, numărul nivelelor, matricile cu soluțiile nivelelor.

### 3.1 Arhitectura



## 4 Mod Utilizare

Clientul își va introduce username-ul în aplicație, acesta va fi verificat și adăugat în leaderboard în cazul în care nu va fi găsit. Se va genera o tablă de joc, iar clientul va trebui să-și facă mutarea introducând de la tastatură pozițiile ce doresc a fi modificate în matrice (e.g. a6-*î*a8). Acestea vor fi verificate, iar în funcție de rezultat, tablă de joc va fi redesenată. În cazul în care mișcarea a fost validă, se va verifica dacă nivelul este terminat, caz în care scorul va fi actualizat și se va genera următoarea tablă pentru următorul nivel, iar dacă nivelul nu este terminat, clientul va putea să-și facă următoarea mișcare, până la finalizarea nivelului.

## 5 referințe

<https://profs.info.uaic.ro/~computernetworks/cursullaboratorul.php>  
[https://lucid.app/lucidchart/d71c83d3-2418-434e-9214-bd4e2d09533d/edit?beaconFlowId=CC9C9534E3F0AE9D&invitationId=inv\\_563d59c5-090a-4d80-add3-36786540c2f2&page=0\\_0#](https://lucid.app/lucidchart/d71c83d3-2418-434e-9214-bd4e2d09533d/edit?beaconFlowId=CC9C9534E3F0AE9D&invitationId=inv_563d59c5-090a-4d80-add3-36786540c2f2&page=0_0#)  
<https://www.cs.dartmouth.edu/~campbell/cs50/socketprogramming.html#:~:text=An%20iterative%20server%20iterates%20through,child%20process%20for%20each%20client.>  
<https://www.binarytides.com/socket-programming-c-linux-tutorial/>

<https://fiimaterials.valentinstamate.com/#materials>