

# MingMangS

Roca Andreea - 2A3

December 8, 2021

## 1 Introducere

Proiectul "MingMangS" presupune implementarea unui joc de strategie. Eu consider ca mintea umana trebuie tinuta continuu activa, astfel jocurile de strategie pot forma inca de la varste fragede mintea unui copil, pana la varste inaintate cand ajuta varstnicii sa isi mentina mintea activa, acestea fiind si un foarte bun mod de a-ti petrece timpul liber cu prietenii, familia, sau chiar jucand aceste tipuri de jocuri online.

Chiar eu de mica am fost pasionata de aceste tipuri de jocuri, mereu incercand sa aflu fiecare strategie de joc pentru a ajunge sa fiu din ce in ce mai buna, si consider ca asta mi-a antrenat memoria pentru a vedea in proiectele si problemele pe care le fac si in ziua de azi fiecare particica care ar putea da eroare.

Astfel am ales proiectul "MingMangS", un joc nou pentru mine, de care nu am mai auzit, dar astfel imi starneste si mai tare interesul.

## 2 Tehnologii utilizate

In implementare am ales sa folosesc un server TCP. Transmission Control Protocol (TCP) este un protocol folosit de obicei de aplicatii care au nevoie de confirmare de primire a datelor. Efectuează o conectare virtuală full duplex între două puncte terminale, fiecare punct fiind definit de către o adresă IP și de către un port TCP.

Desi serverul UDP este mai rapid (pentru jucatorii rapizi) exista riscul de pierdere a datelor. Serverul TCP asigura transmiterea corecta a datelor, astfel nici o mutare a jucatorilor nu se pierde.

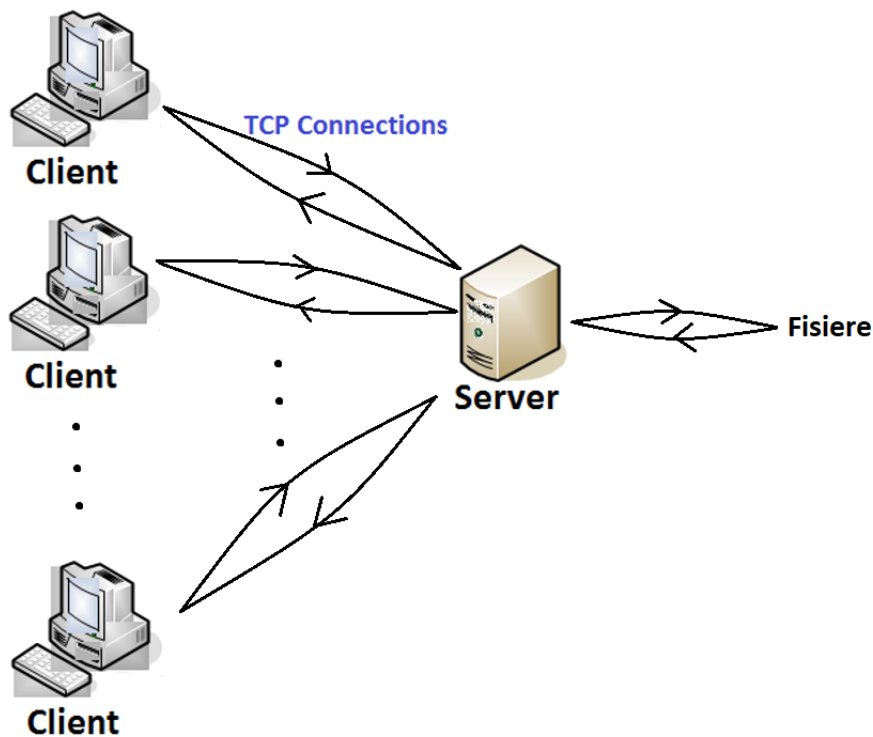
In majoritatea jocurilor conteaza rapiditatea, insa in "boardgames" in general are o mai mare importanta strategia pe care fiecare jucator alege sa o aplice, ordinea in care fiecare mutare este trimisa si precizia cu care acestea sunt facute iar serverul TCP poate asigura toate acestea.

## 3 Arhitectura aplicatiei

Arhitectura client -> server -> fisier text. Clientul va trimite comanda la server, iar acesta din urma va modifica sau va citi informatii din fisierul text daca este necesar.

Spre exemplu la inregistrarea unui utilizator nou, acesta va trimite username-ul si parola aleasa prin client catre server, iar acesta din urma va modifica fisierul text cu informatiile primite din client.

Am ales ca clientul sa nu aiba acces direct la fisierul text pentru pastrarea confidentialitatii tuturor utilizatorilor, astfel nimeni (decat serverul) nu are acces la informatiile (spre ex.: parola) celorlalti utilizatori.



## 4 Detalii de implementare

*Cod relevant particular proiectului*

```
function DESENARETABLA
    for linie = 1; linie <= 8; linie++ do
        for coloana = 1; coloana <= 8; coloana++ do
            tabla[linie][coloana] = 0; //0 reprezinta loc pe tabla unde nu este nici o piesa
        end for
    end for
    for linie = 1; linie <= 8; linie++ do
        tabla[linie][1] = 1; //1 reprezinta jucatorul alb
    end for
    for coloana = 1; coloana <= 7; coloana++ do
        tabla[8][coloana] = 1;
    end for
    for linie = 1; linie <= 8; linie++ do
        tabla[linie][8] = 2; //2 reprezinta jucatorul negru
    end for
    for coloana = 2; coloana <= 8; coloana++ do
        tabla[1][coloana] = 2;
    end for
end function
```

Functia "desenaretabla" va alocă o matrice de 8x8 pe care o va umple la început cu "0" după care va așeza pe marginea din partea de jos și pe cea din partea stângă a matricii piesele jucătorului alb (notate cu "1"), după care va așeza pe marginea din partea de sus și pe cea din partea dreaptă a matricii piesele jucătorului negru (notate cu "2").

```
function JUCATORBLOCAT(numar)
    for linie = 1; linie <= 8; linie++ do
        for coloana = 1; coloana <= 8; coloana++ do
```

```

        if tabla[linie][coloana] == numar then
            if tabla[linie][coloana-1] == 0 || tabla[linie][coloana+1] == 0 || tabla[linie-1][coloana]
== 0 || tabla[linie+1][coloana] == 0 then
                return 1; //exista o mutare posibila si jocul poate continua
            end if
        end if
    end for
end for
return 0; //jucatorul cu numarul "numar" nu mai poate face nici o mutare deci jocul s-a incheiat
end function

```

Functia "jucatorblocat" va primi ca parametrii jucatorul al carui rand este sa mute piesa, urmand sa verifice pentru fiecare piesa a acestui jucator daca are spatii libere in jurul ei, si daca s-a gasit o piesa cu macar un spatiu liber langa ea atunci functia se va incheia. Daca nu a fost gasita o astfel de piesa functia va trimite valoarea 0, ceea ce inseamna ca toate piesele jucatorului sunt blocate deci jocul se va incheia.

```

function INSUFICIENTEPIESE(numar)
    int piese;
    piese = 0;
    for linie = 1; linie <= 8; linie++ do
        for coloana = 1; coloana <= 8; coloana++ do
            if tabla[linie][coloana] == numar then
                piese ++;
            end if
        end for
    end for
    if piese <= 1 then
        return 0; //jucatorul cu numarul "numar" nu mai are suficiente piese pentru a continua jocul
    end if
    return 1; //jucatorul are suficiente piese pentru a continua jocul
end function

```

Functia "insuficientepiese" va primi ca parametrii jucatorul al carui rand este sa mute piesa si va numara pentru acest jucator cate piese mai are disponibile pe tabla. Daca numarul pieselor pe care le mai are este mai mic sau egal cu 1, atunci acest jucator va pierde jocul deoarece nu mai are suficiente piese pentru a mai juca.

### ***Scenarii de utilizare***

"clientul1" se autentifica, serverul verifica daca informatiile pe care acesta le-a trimis sunt corecte si il logheaza, urmand ca "clientul1" sa astepte un alt utilizator sa se conecteze pentru a putea incepe jocul. "clientul2" se autentifica si astfel, serverul va trimite la ambii jucatori tabla de joc.

La fiecare mutare a fiecarui jucator se va verifica validitatea mutarii, (daca piesa este al celui jucator, daca aceasta este blocata, etc.) daca urmeaza sa mute corect, deoarece jucatori se pot muta doar in sus, jos, stanga, dreapta oricate spatii (asemenea unui turn in "sah") si daca acel jucator se afla in una din cele doua posibilitati de incheiere de joc (nu mai are suficiente piese sau toate piesele acestui jucator sunt blocate).

In cazul in care un client va scire comanda "incheie joc" va pierde, urmand ca jucatorul care a ramas in joc sa castige acea runda.

Cand un joc se incheie, serverul va scrie in fisierele pentru clasamente informatii despre acel joc (numarul de piese eliminate, numarul de mutari etc.).

## **5 Concluzii**

In final imi doresc ca acest proiect va avea o implementare foarte clara, cu reguli bine definite si pasi foarte clari, ca orice persoana de orice varsta sa il poata accesa si sa poata juca fara a intampina orice tip de problema.

In final sper ca serverul va reusi sa accepte oricat de multi clienti si va reusi sa ruleze oricat de multe jocuri de-odata, toate acestea fara a se bloca pentru a nu deranja utilizatorii in timpul jocului.

Pe viitor imi doresc ca voi reusi sa implementez un algoritm inteligent pentru ca clientii sa poata juca cu calculatorul, pentru a nu fii nevoiti sa astepte conectarea unui alt client pentru a incepe jocul, deoarece asteptarea poate fi foarte ingelungata, acestia ajungand sa se deconecteze.

Sper ca, pe viitor, voi reusi sa implementez o interfata grafica atractiva pentru a atrage mai multi clienti si astfel experienta din timpul jocului sa fie cat mai placuta.

## References

- [1] TCP protocol specification <https://datatracker.ietf.org/doc/html/rfc793>. Last accessed 8 Dec 2021
- [2] <https://www.quora.com/Which-protocol-would-be-best-to-use-for-gaming-PPTP-UDP-or-TCP>. Last accessed 8 Dec 2021
- [3] <http://www.linktionary.com/t/tcp.html>. Last accessed 8 Dec 2021