

# DraughtsC (B)

Antal Cristian Iulian,  
Grupa B4, Anul 2

Facultatea de Informatică, Iași

## Abstract.

Aceasta este documentația proiectului final de la materia Rețele de calculatoare. Proiectul ales de mine este "DraughtsC"(B) care reprezintă partea de client pentru jocul de Dame, realizat în colaborare cu proiectul "DraughtsS". Această lucrare conține o scurtă introducere și motivația pentru care am ales acest proiect. În continuare voi explica alegerea comunicării de tip TCP și alegerea librăriei grafice SDL. Limbajul folosit este C/C++. În final se regăsește un paragraf ce conține mai multe idei ce pot fi implementate și sursele de unde am preluat informații.

## 1 Introducere

Jocul de dame este foarte cunoscut la nivel global, existând multe variante ale acestui joc în diferite zone sau țări. Ideea de a căuta informații despre aceste variante și a le implementa în proiect m-au făcut să îl aleg fix pe acesta. Limbajul folosit este C/C++.

Astfel, jocul ce urmează a fi implementat va conține următoarele variante de joc:

– Multiplayer:

- *Classic Draughts*

Fiecare jucător are 12 pioni care pot fi mutați pe diagonală doar în față. Când un pion ajunge pe prima linie a adversarului se transformă în damă, care se va putea muta și înainte și înapoi. Pentru a elimina pionul sau dama adversarului trebuie ca pionul/dama să fie pe diagonală iar următoarea celulă de pe diagonală să fie liberă. Jocul se termină atunci când unul dintre jucători rămâne fără piese sau nu mai poate muta nicio piesă (este blocat).

- *Losing Draughts*

Aceleași reguli ca la jocul clasic de dame doar că cel care rămâne fără piese câștigă, astfel strategia de joc se schimbă devenind mult mai interesant.

## 2 Tehnologiile utilizate

Jocul de dame se realizează pe baza protocolului TCP. Acest protocol are la bază conexiuni între mai multe programe care comunică tot timpul.

Am ales protocolul TCP și nu UDP deoarece protocolul UDP nu se bazează pe conexiuni, acesta urmărește doar trimiterea pachetelor de la o sursă către altă sursă nefiind sigur faptul că acestea ajung la destinație. Viteza transmiterii pachetelor este factorul ce face acest protocol folosit mai ales pentru programe ce necesită informații real-time cum ar fi jocurile de tip "shooter", transmiterea de videoclipuri etc. La jocul de Dame este mai importantă primirea informațiilor decât viteza cu care sunt transmise datele, de aceea protocolul TCP este potrivit pentru acest proiect.

**TCP - Transmission Control Protocol** este un protocol sigur pentru realizarea unei conexiuni între clienți și server. Transmiterea de mesaje între aceste instanțe este una sigură deoarece mereu se așteaptă confirmarea primirii datelor. Pentru jocul de Dame acest protocol aduce următoarele avantaje:

- o conexiune stabilă și permanentă între clienți și server (până la închiderea ei);
- ușurința și fiabilitatea comunicării între aceste instanțe.

Jocul de Dame implică un schimb de mesaje în permanență între clienți și server. Primele cerințe către server de la client vor fi de login/sign up, după acestea toate cererile trimise de client vor fi rezolvate de server, acesta trimițând mesaj înapoi către client care va afișa informațiile într-un mod atractiv (în cazul afișării clasamentului) sau va realiza o acțiune conform răspunsului de la server (o mutare a unei piese în cadrul unui joc). Când jucătorul va alege opțiunea de "Exit" acesta va opri comunicarea cu serverul.

Nevoia de server TCP concurent vine de la faptul că acesta trebuie să servească mai mulți clienți simultan și să ofere posibilitatea de a exista mai multe jocuri simultan.

**Qt** este un widget toolkit care oferă posibilitatea creării unui GUI (Graphical User Interface) cât mai atractiv pentru a facilita interacțiunile clientului cu serverul. Qt Creator este IDE-ul special creat pentru Qt care facilitează foarte mult programarea. Acesta funcționează pe baza de OOP, având foarte multe clase cu metode proprii ce facilitează foarte mult munca depusă.

Am ales Qt deoarece oferă posibilități și funcționalități multiple mai ales pentru design grafic 2D, este intuitivă și ușor de folosit.

Câteva funcționalități pe care le oferă Qt sunt:

- Video: grafică 2D/3D, poate crea mai multe ferestre, poate fi folosită în combinație cu alte librării (OpenGL);
- Threads: creare de thread API simplu, stocare locală a unui thread API simplu, semafoare etc.;
- Ceasuri: creează ceasuri ce rulează pe un thread separat, numără miliseconde, poate aștepta un anumit număr de miliseconde etc.;
- Și extrem de multe alte clase pentru diverse elemente de GUI.

### 3 Arhitectura aplicatiei

Pentru o bună organizare a funcționalităților din client se vor folosi principii de OOP, deoarece cu ajutorul acestora modificarea, prelucrarea și organizarea datelor va fi mult mai clară.

Cum am specificat și la punctul anterior, serverul va fi TCP concurent, deci fiecare client în parte se va conecta la server, comunicând cu un proces copil creat pentru a le procesa cerințele.

Diagrama de pe pagina următoare prezintă comunicarea dintre server și clienți.

### 4 Detalii de implementare

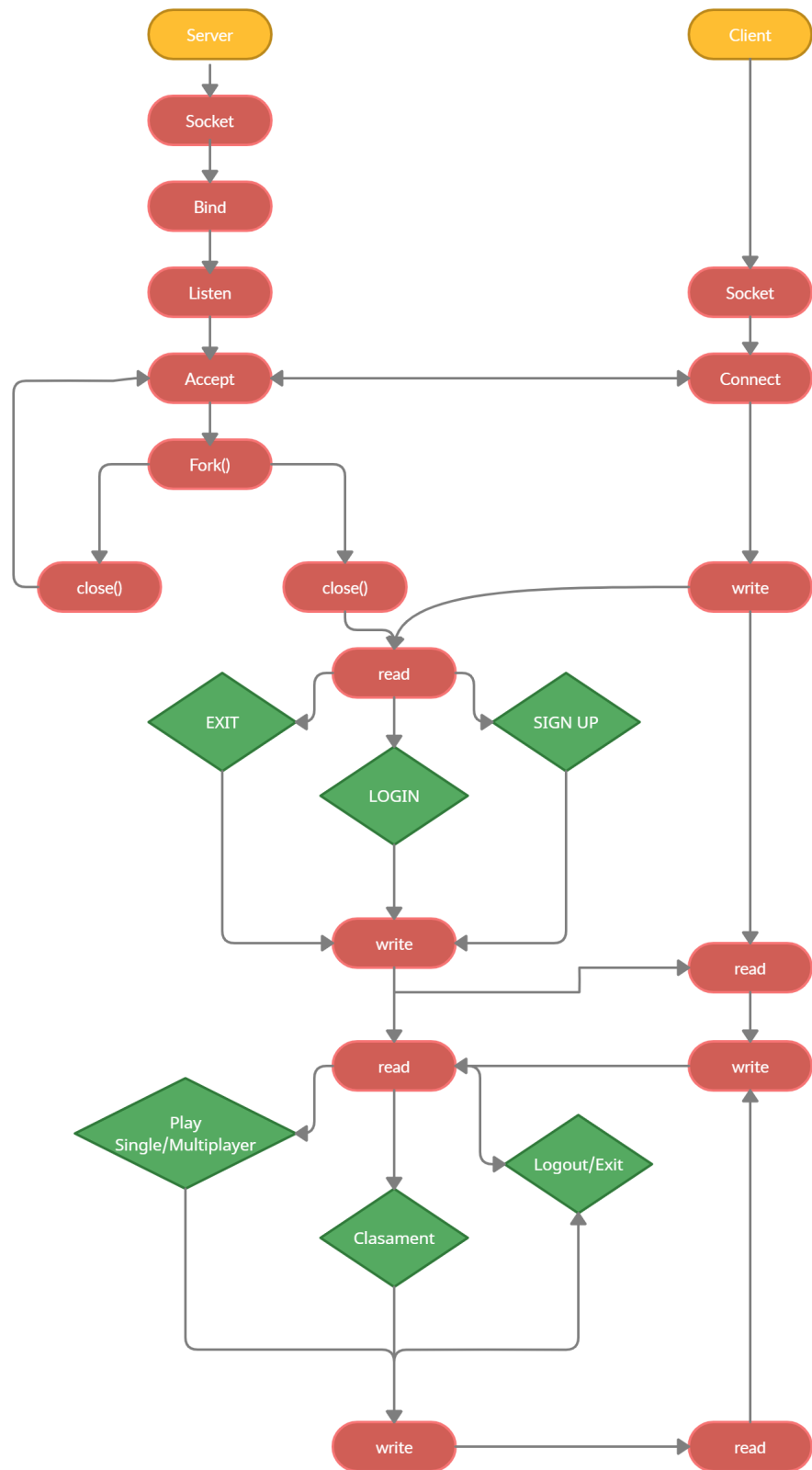
#### 4.1 O parte din clasele implemmentate in client si rolul lor:

- *mainwindow.cpp-mainwindow.h* //clasa pentru prima fereastră.
- *secondwindow.cpp-secondwindow.h* //clasa pentru fereastră de signup
- *thirdwindow.cpp-thirdwindow.h* //clasa pentru fereastră de login
- *mainmenu.cpp-mainmenu.h* //clasa pentru meniul principal
- *ranking.cpp-ranking.h* //clasa pentru pagina de clasament
- *gamewindow.cpp-gamewindow.h* *game.cpp-game.h* //clase pentru jocul clasic de dame
- *lgamewindow.cpp-lgamewindow.h* *lgame.cpp-lgame.h* //clase pentru jocul de losing draughts
- *connection.cpp-connection.h* //clasa cu rolul de a copia descriptorul socket-ului si a-l transmite mai departe in alte clase

#### 4.2 Scenarii de utilizare

Aplicația va avea mai multe ferestre:

- Prima fereastră cu următoarele opțiuni:
  1. *sign up*: când clientul intră pentru prima dată în aplicație, acesta va introduce un username, dacă există deja va primi un mesaj astfel încât să fie nevoit să introducă alt username, dacă este corect acesta va fi trimis la fereastră a doua;
  2. *log in*: se introduc doua username-uri. Dacă se introduce un username inexistent sau acelasi username de doua ori, acesta va primi un mesaj pentru a fi înștiințat, dacă introduce un username valid va fi trimis la a doua fereastră;



3. *exit*: clientul închide aplicația și oprește comunicarea cu serverul.
- A doua fereastră oferă clientului posibilitatea de a alege următoarele opțiuni: *exit*.
    1. *Classic Draughts*: joc clasic de dame, scopul este de a elimina toate piesele inamicului;
    2. *Losing Draughts*: scopul este de a ramane fara piese, in acest caz toata strategia jocului se schimba radical;
    3. *Ranking*: se afiseaza un clasament ce contine primii 10 jucatori si punctajele lor;
    4. *Logout*: clientul este trimis la prima fereastră;
    5. *Exit*: clientul închide aplicația și oprește comunicarea cu serverul.
    6. *Urmatoare implementări prezentate la secțiunea de "Concluzii"*.
  - Jocul propriu zis: acesta are loc într-o fereastră nouă ce conține: tabla și piesele, un text ce indică rândul cărui jucător este de a muta o piesă. La final, pe ecran va fi afișat jucătorul câștigător și un buton ce ne redirectioneaza catre meniul principal. Fiecare joc câștigat valorează 100 de puncte, iar un joc pierdut valorează 25 puncte.

## 5 Concluzii

Fiind un joc destul de simplu, se pot implementa variațiuni cu o multitudine de reguli noi. De asemenea, pentru a crea o interfață și o experiență de utilizare cât mai bună se pot implementa următoarele idei:

- adăugarea a mai multe variațiuni ale jocului de Dame;
- mod de joc de tip *turneu*: se aleg 12 jucători aleatoriu, sunt grupați câte 2 pentru prima rundă. Cei 6 câștigători sunt grupați iar cate 2, cei eliminați primesc 10 puncte pentru participare. Următorii 3 câștigători vor juca fiecare cu fiecare, cel ce va câștiga cele mai multe meciuri, va primi 100 de puncte, locul 2 va primi 75 de puncte iar locul 3 va primi 50 de puncte;
- implementarea autentificării și pe bază de parolă;
- opțiunea de a vizualiza lobby-urile existente și de a alege unul;
- opțiunea de "Friendly Match", unde te poți juca cu un anumit prieten, fără a câștiga puncte;
- opțiunea de a da "Rematch" la final de joc;
- mai multe tipuri de clasamente: cel mai rapid meci, cel mai lung meci, cele mai multe meciuri câștigate, cele mai multe turnee câștigate;

## 6 Bibliografie

1. <https://creately.com/> - crearea diagramei

2. <https://profs.info.uaic.ro/computernetworks/cursulaboratorul.php>
3. <https://profs.info.uaic.ro/georgiana.calancea/laboratories.html>
4. <https://beej.us/guide/bgnet/html/>
5. <https://en.wikipedia.org/wiki/Draughts>
6. <https://www.libsdl.org/>
7. <http://wiki.libsdl.org/Tutorials>