



ChessS

Duca Alexandru(2E1)

Facultatea de Informatica, Iasi

19 Decembrie 2022

ChessS (B)

Sa se conceapa o aplicatie server care pune la dispozitie o tabla de joc si supervizeaza desfasurarea fiecarei partide de sah, actionand ca un punct central la care clientii din retea se conecteaza. Serverul determina momentul in care jocul s-a terminat si anunta castigatorul. Regulile pot fi alese (simplificate) de catre proiectant, cu conditia ca jocul sa fie interactiv.

Cuprins

1	Introducere	2
2	Tehnologiile utilizate	2
3	Arhitectura aplicatiei	2
3.1	Conceptele implicate	2
3.2	Protocol	3
3.3	Diagrama aplicatiei	4
4	Detalii de implementare	5
4.1	Cod relevant particular proiectului	5
4.2	Scenarii de utilizare	7
5	Concluzii	16
6	Bibliografie	17

1 Introducere

Proiectul ChessS(B) presupune crearea unei aplicatii server ce actioneaza ca un punct central de conectare pentru clientii din retea, punand la dispozitie o tabla de joc si supervizand desfasurarea fiecarei partide. Jocul este interactiv si se intituleaza ca fiind un sah feminist(Am decis sa-l numesc asa pentru ca, spre deosebire de jocul clasic, aici regina este cea mai importanta piesa), respectand in mare masura regulile de mutare ale sahului clasic, singurele modificari fiind faptul ca regele poate fi capturat la randul sau si ca pentru a castiga meciul, jucatorii trebuie sa captureze regina adversarului. Serverul determina momentul in care jocul s-a terminat si anunta deznodamantul partidei.

2 Tehnologiile utilizate

Protocolul de transfer folosit este TCP (Transmission Control Protocol). TCP este orientat spre transmiterea corecta a informatiilor, lucru absolut necesar pentru buna desfasurare a unui joc de sah(chiar daca jocul clasic e putin modificat), unde fiecare mutare trebuie luata in vedere cu exactitate. Utilizarea acestui protocol ne asigura de asemenea ordine si fiabilitate.

Precizia este mult mai importanta decat viteza de transfer, tocmai de aceea se foloseste un protocol TCP si nu un protocol UDP.

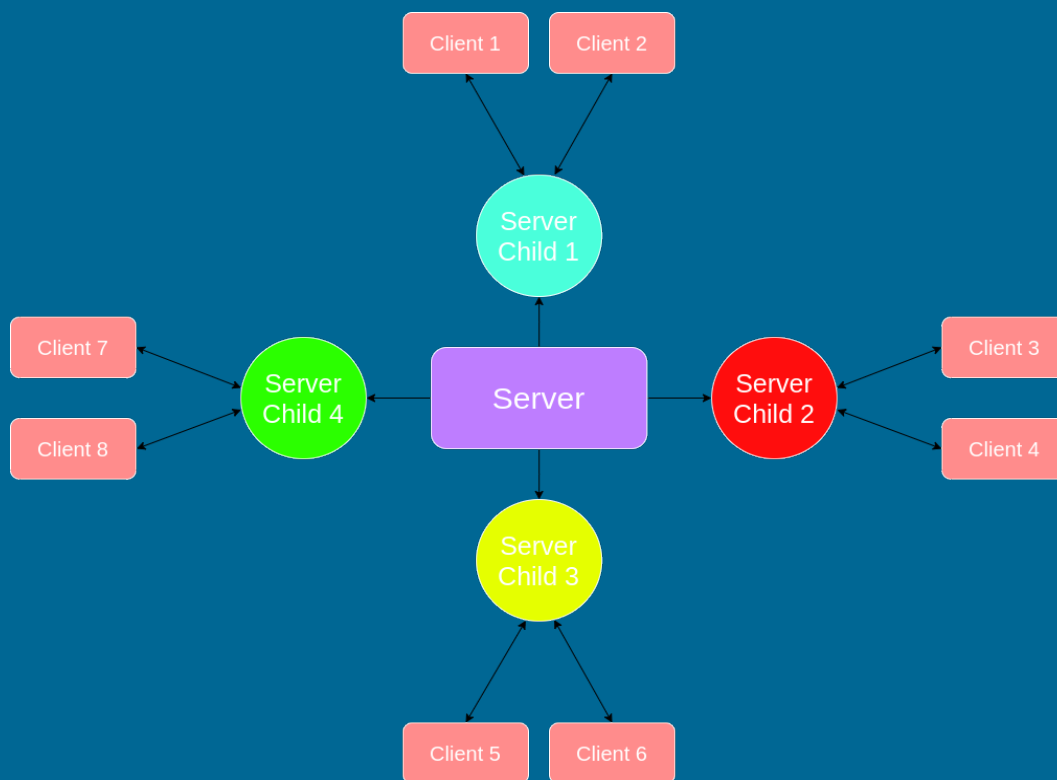
Se utilizeaza TCP concurent, deoarece se doreste conectarea mai multor utilizatori si desfasurarea mai multor partide de sah in acelasi timp.

3 Arhitectura aplicatiei

3.1 Conceptele implicate

Serverul creeaza mai intai un socket si il leaga de o adresa si de un port cunoscut pentru ca clientii sa se poata conecta. Dupa aceea, serverul intra intr-o bucla conceputa pentru a asculta clientii in mod concurent. Cand serverul a acceptat cu succes doi clienti creeaza un proces copil nou ce va juca rol de camera, unde se va desfasura jocul intre cei doi, intorcandu-se apoi la inceputul buclei pentru a asculta noi clienti si a initia noi partide. In momentul in care jocul s-a incheiat intr-o camera, conexiunea celor doi clienti se inchide automat si se face exit si in procesul copil pentru a nu avea procese zombie.

3.2 Protocol



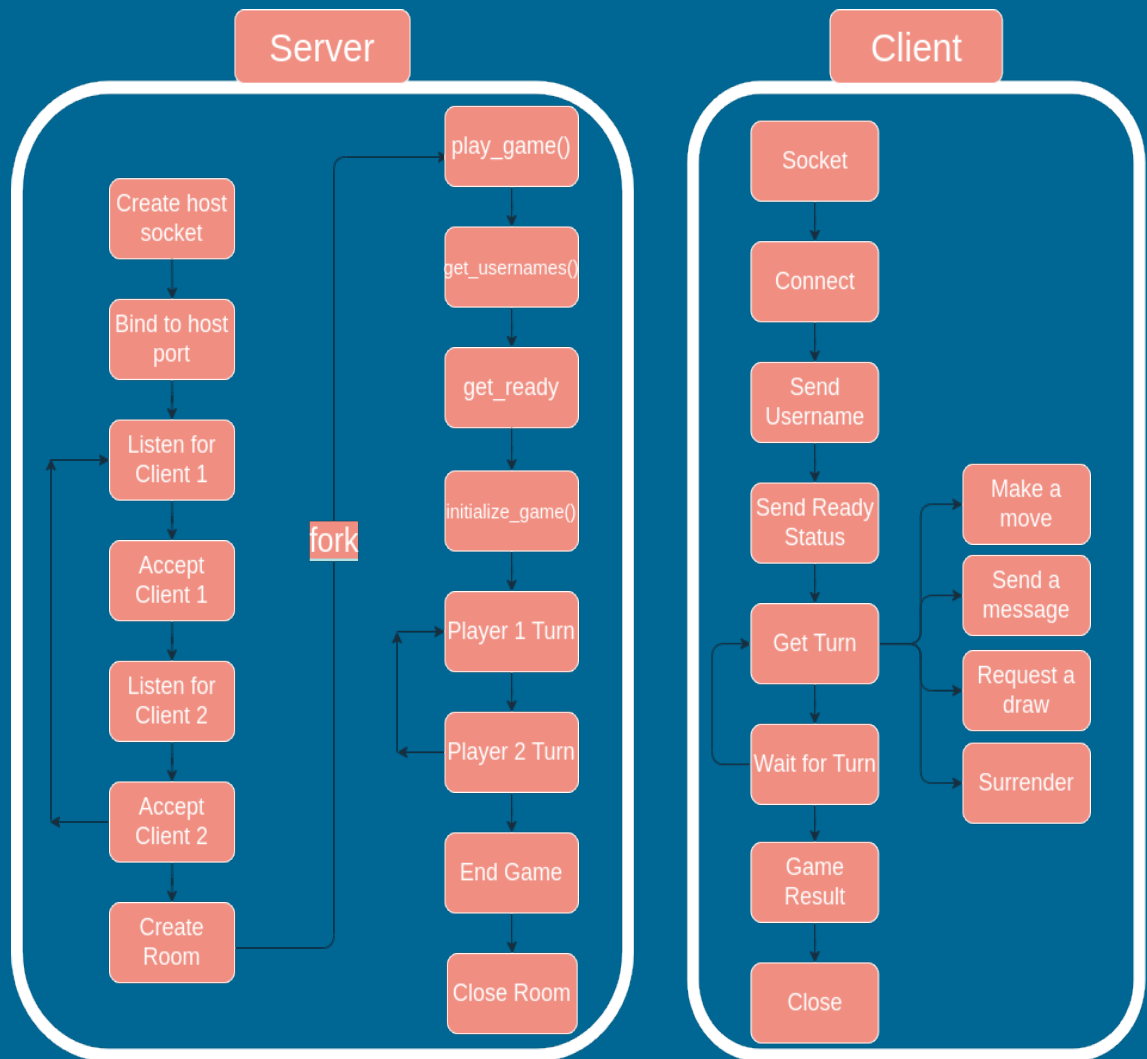
Pentru a se conecta la server, clientii trebuie sa specifice drept parametrii adresa IP si port-ul: `./client [IP_ADRESS] [PORT]`. Adresa IP a serverului este 0.0.0, iar port-ul este 2024.

Serverul, fiind unul concurent, accepta conectarea a mai multi clienti si desfasurarea concomitenta a mai multor partide. Dupa ce se conecteaza doi clienti, serverul creeaza un proces copil pentru ei. Acest lucru se intampla de fiecare data cand un numar par de clienti sunt conectati. Procesele copil nu comunica deloc cu procesul parinte. Ele supervizeaza desfasurarea partidei pana la final, dupa care se inchid.

Dupa ce sunt repartizati catre un proces copil, cei doi clienti nu ajung niciodata sa comunice intre ei in mod direct. Asa cum se poate vedea si in diagrama, clientii pot doar sa primeasca de la procesul copil la care au fost asignati si sa-i trimita inpoi, efectuand doar operatii de `recv()` si `send()`.

Pe tot parcursul jocului, procesul copil server comunica pe rand cu cei doi clienti pe care ii are in supervizare, triminandu-le si primind diferite mesaje de tip text de diferite dimensiuni, in functie de optiunile alese.

3.3 Diagrama aplicatiei



4 Detalii de implementare

4.1 Cod relevant particular proiectului

- **play_game()** - Aceasta functie se apeleaza dupa crearea procesului copil si dicteaza desfasurarea intregului joc. In interiorul ei se apeleaza pe rand functiile: **get_usernames()**, **get_ready()**, **initialize_game()**. Apoi se intra intr-o bucla in care randul celor doi jucatori se schimba dupa fiecare mutare. Atunci cand e randul lor, jucatorii sunt informati in legatura cu optiunile pe care le au. Iesirea din aceasta bucla este echivalenta cu incheierea partidei.
- **get_usernames()** - In momentul in care se apeleaza, cei doi jucatori primesc simultan de la server instructiunea de a-si alege un username care sa contina cel putin un caracter si de a apasa tasta ENTER. In caz contrar, cei doi primesc un mesaj de eroare in care li se spune ca e nevoie ca username-ul lor sa aiba cel putin un caracter si li se ofera sansa de a-si seta din nou un username. Eroare va persista pana cand jucatorul va introduce un username valid.
- **get_ready()** - Dupa ce jucatorii si-au setat un username, acestia primesc de la server instructiunea de a apasa tasta ENTER in momentul in care sunt pregatiti pentru inceperea jocului. In urma apasarii, vor primi un mesaj de confirmare ca sunt pregatiti.
- **initialize_game()** - Aceasta functie primeste drept unul dintre parametri functia **first_in_alphabetical_order()** ce compara username-urile celor 2 jucatori si decide care dintre ei va avea prima mutare. Va incepe cel cu username-ul mai inainte in ordine alfabetica. Astfel, jucatorii vor fi informati in legatura cu: ceea ce trebuie sa faca pentru a castiga meciul(capturarea reginei adversarului), reprezentare grafica a tablei de sah impreuna cu explicarea notatiilor folosite in aceasta reprezentare si care este culoarea pieselor cu care vor juca.
- **verify()** - Este apelata in momentul in care un jucator doreste sa execute o mutare pentru a verifica daca mutarea este sau nu posibila. Pentru a face acest lucru, se foloseste de multe alte functii, fiecare verificand o anumita chestie si returnand un cod de eroare: **length_check()**, **valid_piece_check()**, **position_check()**, **dashes_check()**, **basic_check()**, **rock_move_check()**, **knight_move_check()**, **bishop_move_check()**, **king_move_check()**, **queen_move_check()**, **pawn_move_check()**.
- **move_errors()** - Aceasta functie primeste un cod de eroare si returneaza mesajul de eroare pentru codul respectiv.
- **make_a_move()** - Se poate apela cand este randul unui jucator pentru a-si executa mutarea. Apelarea acestei functii este obligatorie pentru a se putea continua jocul si a i se oferi randul adversarului. Atunci cand alege

sa isi faca mutarea, jucatorul este informat in legatura cu sintaxa pe care trebuie sa o foloseasca. Daca functia **verify()** decide ca mutarea este gresita, se foloseste functia **move_errors()** pentru a i se trimite jucatorului mesajul de eroare corespunzator greselii pe care a facut-o si i se ofera din nou sansa de a introduce o mutare. Acest lucru se intampla pana cand jucatorul introduce o mutare valida. Daca mutarea este corecta, se actualizeaza tabla de sah si li se afiseaza din nou amandurora reprezentarea tablei actualizate, adversarul fiind informat exact si in legatura cu mutarea efectuata. In final, randul este acordat celuilalt jucator.

- **send_a_message()** - Poate fi apelata doar atunci cand este randul jucatorului pentru a-i trimite un mesaj adversarului. Actiunea poate fi repetata de un numar nelimitat de ori de catre jucator atunci cand este randul lui.
- **request_a_draw()** - Poate fi apelata o singura data de catre fiecare jucator pe parcursul jocului in momentul in care se doreste solicitarea unei remize. Oponentului i se da dreptul de a accepta, caz in care jocul este considerat egal sau de a refuza, caz in care jocul continua.
- **surrender()** - Poate fi apelata daca un jucator nu mai doreste sa continue meciul. Aceasta functie ii permite sa se deconecteze de la server. In acest caz, adversarul este declarat castigator.
- **disconnected()**, **disconnected_nouusername()**, **disconnected_ready()**, **disconnected_initialize()** - Aceste functii sunt concepute ca in cazul care unul dintre utilizatori se deconecteaza in mod neasteptat de la server, celalalt jucator sa fie anuntat si meciul sa se incheie.

4.2 Scenarii de utilizare

Cand s-au conectat doi clienti, acestia sunt grupati impreuna si primesc simultan de la server instructiunea de a-si alege un username care sa contina cel putin un caracter si de a apasa tasta ENTER.

```
alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-proiect$ ./client 0 2024
Connected to the server.

Choose an username(at least one character) and press ENTER: 
```

In cazul in care unul din clienti apasa tasta ENTER fara a introduce macar un caracter, acesta primeste un mesaj de eroare in care i se spune ca e nevoie ca username-ul sa aiba cel putin un caracter si i se ofera din nou sansa de a-si seta un username.

```
alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-proiect$ ./client 0 2024
Connected to the server.

Choose an username(at least one character) and press ENTER:

You have to use at least one character:
Choose your username: 
```

Dupa ce amandoi jucatorii si-au setat un username, fiecare este instiintat de numele adversarului si li se solicita sa apese tasta ENTER atunci cand sunt pregatiti pentru inceperea jocului.

```
=====
Your opponent is Paul.
Press ENTER when you're ready
```

In urma apasarii, vor primi un mesaj de confirmare ca sunt pregatiti.

```
=====
Your opponent is Paul.
Press ENTER when you're ready
READY!
=====
```

Cand amandoi jucatorii au decis ca sunt pregatiti, jocul incepe si amandoi primesc urmatoarele informatii in legatura cu:

- Modalitatea de a castiga meciul

```
=====
START GAME!

This is a feminist chess. To win the game, you have to capture your opponent's Queen.
GOOD LUCK!
```

- Reprezentare grafica a tablei de sah impreuna cu explicarea notatiilor folosite

```
The notations are:
- R for a Rook
- N for a Knight
- B for a Bishop
- K for a King
- Q for a Queen
- P for a Pawn

To distinguish the colors, the white pieces will be preceded by 'w' and the black pieces will be preceded by 'b'.
```

	a	b	c	d	e	f	g	h	
1	bR	bN	bB	bQ	bK	bB	bN	bR	1
2	bP	bP	bP	bP	bP	bP	bP	bP	2
3	---	---	---	---	---	---	---	---	3
4	---	---	---	---	---	---	---	---	4
5	---	---	---	---	---	---	---	---	5
6	---	---	---	---	---	---	---	---	6
7	wP	wP	wP	wP	wP	wP	wP	wP	7
8	wR	wN	wB	wQ	wK	wB	wN	wR	8
	a	b	c	d	e	f	g	h	

- Culoarea pieselor cu care vor juca

```
You have the WHITE pieces (those starting with 'w').

You have the BLACK pieces (those starting with 'b').
```

Daca nu este randul lui, jucatorul trebuie sa astepte pana cand adversarul isi face mutarea.

```
=====

It's not your turn, you have to wait until your opponent makes his move.

█
```

Daca este randul lui, atunci jucatorul are la dispozitie urmatoarele optiuni: sa execute o mutare, sa-i trimita un mesaj adversarului, sa solicite o remiza si sa renunte. Pentru a selecta o optiune e necesar sa introduca numarul ei si sa apese

tasta ENTER.

```
=====
It's your turn. What do you want to do?
1.Make a move
2.Send a message to your opponent
3.Request a draw
4.Surrender

To choose an option, just write its number and press ENTER.
█
```

Daca jucatorul a scris altceva decat numarul celor 4 optiuni disponibile, atunci acestuia i se transmite faptul ca optiunea nu exista si ii este din nou oferita sansa de a alege o optiune. Acest lucru se intampla la nesfarsit pana ce utilizatorul alege o optiune valida.

```
6

=====
Option does not exist. Please choose an available option.
1.Make a move
2.Send a message to your opponent
3.Request a draw
4.Surrender

To choose an option, just write its number and press ENTER.
█
```

Daca jucatorul a ales sa faca o mutare, va fi informat cu privire la sintaxa pe care trebuie sa o respecte pentru a efectua mutarea cu succes.

```
1

=====
Each square on a chess board is given by a coordinate.
To make a move, you have to specify:
    -the piece you want to move;
    -the coordinate from which you are moving;
    -the destination coordinate.
ex: bR-e3-e5
█
```

In cazul in care jucatorul a introdus o mutare gresita, i se trimite mesajul de eroare corespunzator greselii pe care a facut-o si i se ofera din nou sansa de a introduce o mutare. Acest lucru se intampla pana cand jucatorul introduce o mutare valida.

```
sadjkjas

=====

Invalid move.
The piece you selected does not exist.
Please enter a correct move:

bP-a2-a4

=====

Invalid move.
You can't move your opponent's pieces!
Please enter a correct move:

wR-a8-a7

=====

Invalid move.
You may not kill your own pieces.
Please enter a correct move:

█
```

Daca jucatorul a efectuat o mutare corecta, se actualizeaza tabla de sah si amandoi jucatorii primesc reprezentarea tablei actualizate, adversarul fiind informat exact si in legatura cu mutarea efectuata. In urma mutarii, randul este acordat celuiilalt jucator.

```
wP-c7-c5
  a  b  c  d  e  f  g  h
1  bR bN bB bQ bK bB bN bR 1
2  bP bP bP bP bP bP bP bP 2
3  --- --- --- --- --- --- --- 3
4  --- --- --- --- --- --- --- 4
5  --- --- wP --- --- --- --- 5
6  --- --- --- --- --- --- --- 6
7  wP wP --- wP wP wP wP wP 7
8  wR wN wB wQ wK wB wN bR 8
  a  b  c  d  e  f  g  h

=====
```

Daca pe masura desfasurarii partidei, unul dintre cei doi jucatori captureaza regina adversarului jocul se incheie.

wP-g6-h5

	a	b	c	d	e	f	g	h	
1	__	bN	bB	bQ	bK	bB	__	bR	1
2	__	__	bP	__	bP	bP	bP	bP	2
3	__	bP	__	__	__	__	__	__	3
4	bR	__	__	bP	__	wP	__	__	4
5	__	__	__	__	__	__	__	wP	5
6	__	__	wP	wP	__	__	__	__	6
7	wP	__	__	__	wP	bN	__	__	7
8	wR	wN	wB	wQ	wK	wB	wN	__	8
	a	b	c	d	e	f	g	h	

La o mutare de capturarea reginei si implicit de finalul jocului

In loc sa primeasca doar reprezentarea tablei, ca in urma executarii unei mutari obisnuite, de aceasta data ei sunt anuntati direct ca regina a fost capturata.

=====																	=====
	You captured your opponent's Queen!																
	a	b	c	d	e	f	g	h									
1	__	bN	bB	bQ	bK	bB	__	bR	1								
2	__	__	bP	__	bP	bP	bP	bP	2								
3	__	bP	__	__	__	__	__	__	3								
4	bR	__	__	bP	__	wP	__	__	4								
5	__	__	__	__	__	__	__	wP	5								
6	__	__	wP	wP	__	__	__	__	6								
7	wP	__	__	__	wP	__	__	__	7								
8	wR	wN	wB	bN	wK	wB	wN	__	8								
	a	b	c	d	e	f	g	h									
	Your Queen has been captured!																
	a	b	c	d	e	f	g	h									
1	__	bN	bB	bQ	bK	bB	__	bR	1								
2	__	__	bP	__	bP	bP	bP	bP	2								
3	__	bP	__	__	__	__	__	__	3								
4	bR	__	__	bP	__	wP	__	__	4								
5	__	__	__	__	__	__	__	wP	5								
6	__	__	wP	wP	__	__	__	__	6								
7	wP	__	__	__	wP	__	__	__	7								
8	wR	wN	wB	bN	wK	wB	wN	__	8								
	a	b	c	d	e	f	g	h									

Acestia sunt informati cu privire la deznodamantul partidei, primind fie mesaj de invingator, fie de pierzator. In final, conexiunea lor la server este inchisa.

```

=====
GAME OVER
YOU WON! CONGRATULATIONS!
=====

Closing the connection...
alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-project$

```

```

=====
GAME OVER
YOU LOST:( BETTER LUCK NEXT TIME!
=====

Closing the connection...
alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-proiect$ █

```

Daca utilizatorul alege sa trimita un mesaj adversarului, acesta va scrie ceea ce isi doreste sa-i transmita si va apasa tasta ENTER. Actiunea poate fi repetata de un numar nelimitat de ori de catre jucator atunci cand este randul lui.

```

2
=====

Write your message to Paul and press ENTER:
Good luck! Let's have fun!

=====

```

In urma trimiterii, adversarul primeste mesajul si este informat ca trebuie sa astepte pana vine randul lui pentru a-i raspunde.

```

=====

Message received from Alex:
Good luck! Let's have fun!
You have to wait for your turn to reply.

█

```

Daca un jucator simte ca jocul nu se indreapta in nicio directie poate alege optiunea de a-i solicita adversarului o remiza amiabila. Aceasta optiune poate fi folosita doar o singura data pe durata partidei de catre fiecare participant.

```

3
=====

You can request a draw once per game. Are you sure you want to request a draw?
Type 'y' and press ENTER to confirm.
OR
Type anything else and press ENTER to go back to the options.

```

Daca in momentul in care i se cere confirmarea, jucatorul se razgandeste poate introduce orice altceva si sa apese ENTER si ii vor fi din nou afisate optiunile.

Daca isi doresti in continuare sa solicite remiza el introduce 'y' si apasa ENTER si apoi va fi nevoit sa astepte un raspuns din partea adversarului la cererea sa.

```
y
Waiting for the opponent to give an answer...

=====

=====

Alex has request a draw.
1.Accept
2.Decline

To choose an option, just write its number and press ENTER.

□
```

Dacă adversarul refuză, jocul continuă și jucătorului care a cerut remiza îi apar din nou opțiunile. Nu va mai putea pe durata aceluși meci să solicite din nou o remiza.

```
2
You declined Alex's request. The game continues.

=====
Paul has declined your request. The game continues.
```

Daca adversarul accepta, partida se incheie cu un rezultat de egalitate, amandoi primesc acelasi mesaj de final, iar conexiunea la server se inchide.

```

#####
GAME OVER!
DRAW BY MUTUAL AGREEMENT.
GOOD GAME!
#####

Closing the connection...

alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-projects$

```

Daca utilizatorul doreste sa se declare invins si sa paraseasca meciul imediat, el poate alege optiunea de a renunta atunci cand este randul lui.

```
4
=====

Are you sure you want to surrender?
Type 'y' to confirm.
Press any other button to go back to the options.
█
```

In cazul in care jucatorul se razgandeste si apasa orice alt buton, acesta primeste un mesaj si i se afiseaza din nou optiunile pentru a-si continua randul. Va putea insa se intoarca la aceasta optiune oricand va dori.

```
dsf
=====

It looks like you changed your mind!
NEVER GIVE UP!
```

Daca insa acesta este hotarat si scrie 'y' pentru confirmare, atunci meciul se incheie, adversarul este declarat castigator, jucatorii fiind informati in legatura cu aceste lucruri, iar in final conexiunea lor la server este inchisa.

```
=====
Are you sure you want to surrender?
Type 'y' to confirm.
Press any other button to go back to the options.
y

=====

GAME OVER
YOU SURRENDERED:( BETTER LUCK NEXT TIME!

=====

Closing the connection...

alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-projects$
```

```

=====
GAME OVER
YOUR OPPONENT SURRENDERED. YOU WON!
=====

Closing the connection...

alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-projects$

```

În cazul în care conexiunea unuia dintre clienți se închide după ce aceștia au fost grupați împreună, dar înainte ca partida să înceapă, clientul rămas conectat primește un mesaj în care este anunțat în legătură cu acest lucru și informat că va trebui să se conecteze din nou la server pentru a putea găsi un alt meci, după care se închide și conexiunea lui la server.

```
=====
Alex disconnected.
You'll need to reconnect to the server to find a new match.

Closing the connection...
alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-proiect$
```

Dacă conexiunea unuia dintre clienți se închide după începerea jocului, clientul rămas conectat va fi declarat câștigător, va fi primit un mesaj corespunzător și conexiunea lui la server va fi și ea închisă.

```
=====
GAME OVER!
Alex disconnected.
YOU WON!

Closing the connection...
alex@alex-VivoBook-ASUSLaptop-X509FJ-X509FJ:~/Documents/Chess-proiect$
```

5 Concluzii

Proiectul reuseste sa atinga toate obiectivele stabilite, punand la dispozitie o aplicatie server concurrent, ce permite conectarea mai multor utilizatori si desfasurarea in mod simultan a mai multor partide interactive de sah feminist(o varianta simplificata a jocului clasic de sah). Aplicatia le pune la dispozitie jucatorilor, de asemenea si o reprezentare grafica a tablei. La finalul partidei, serverul anunta un castigator, un rezultat de egalitate sau nu anunta un verdict in cazul unui meci terminat brusc din cauza unei deconectari a unuia dintre clienti.

Cum ar putea fi imbunatatit proiectul?

- Crearea unui interfate grafice pentru o mai buna desfasurare a jocului.
- Adaugarea posibilitatii jucatorilor de a schimba de mesaje, de a solicita o remiza sau de a renunta si atunci cand nu este randul lor.
- Optiunea crearii unui cont la care utilizatorii sa se autentifice de fiecare data cand se conecteaza la server.
- Oferirea posibilitatii clientilor de a se conecta la server si de a li se oferi un meniu din care sa poata alege ceea ce doresc sa faca: sa desfasoare un meci, sa priveasca un meci deja aflat in desfasurare din postura de spectatori, urmarirea istoricului meciurilor desfasurate(in cazul implementarii optiunii de creare de cont).
- Oferirea optiunii de rematch celor doi clienti dupa terminarea fiecarei partide si pastrarea unui scor intre cei doi.

6 Bibliografie

[1] Computer Networks - Faculty of Computer Science Website

- https://profs.info.uaic.ro/~computernetworks/files/5rc_ProgramareaInReteaI_en.pdf
- https://profs.info.uaic.ro/~computernetworks/files/6rc_ProgramareaInReteaII_En.pdf
- https://profs.info.uaic.ro/~computernetworks/files/7rc_ProgramareaInReteaIII_En.pdf
- <https://profs.info.uaic.ro/~computernetworks/files/NetEx/S9/servTcpCSEL.c>
- <https://profs.info.uaic.ro/~computernetworks/files/NetEx/S9/cliTcp.c>

[2] Chess Rules

- <https://www.chess.com/terms/chess-pieces>
- <https://www.chess.com/forum/view/general/draw-offer-etiquette>
- <https://www.chessmatec.com/chess-rules-for-kids-draw>
- https://en.wikibooks.org/wiki/Chess/Arranging_The_Board
- https://en.wikibooks.org/wiki/Chess/Notating_The_Game

[3] Others

- <https://stackoverflow.com/questions/2770235/c-child-read-giving-resource-temporarily-unavailable>
- http://www.qnx.com/developers/docs/qnxcar2/index.jsp?topic=/com.qnx.doc.pps.developer%2Ftopic%2Fsubscribe_blocking.html
- <https://stackoverflow.com/questions/283375/detecting-tcp-client-disconnect>
- <https://www.tutorialspoint.com/how-do-i-catch-a-ctrlplusc-event-in-cplusplus>
- <https://stackoverflow.com/questions/17318886/fflush-is-not-working-in-linux/2388500823885008>
- <https://stackoverflow.com/questions/21537662/epipe-error-does-not-appear>