

Pheasant Project

☞Bobu Dragos-Andrei

7 decembrie 2022

Rezumat

Prezentarea tehnologiilor utilizate, a arhitecturii și a detaliilor de implementare a proiectului ChessC (B).

Cuprins

1	Introducere	2
2	Game Flow	2
3	Tehnologii utilizate	2
4	Arhitectura si detalii de implementare	4
4.1	Funcțiile grafice	5
4.1.1	drawLevel()	5
4.1.2	drawMove()	5
4.1.3	undoLastMove()	5
4.2	Username	5
4.3	Scorul	5
4.4	Verificarea mutării	5
5	Design	6
6	Utilizare	6
7	Concluzii	6

1 Introducere

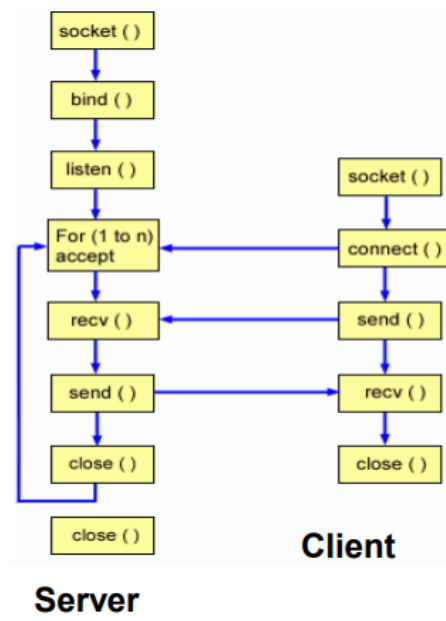
Scopul acestui proiect este de a dezvolta o aplicație client/server care să pună la dispoziția utilizatorilor un mediu de desfășurare al jocului Fazan.

2 Game Flow

1. Jucatorul 1 alege o litera.
 - (a) Serverul trimite litera aleasa Jucatorului 2
 - (b) Jucatorul 2 primește litera aleasa
2. Jucatorul 2 introduce un cuvânt valid care să înceapă cu litera aleasa precedent.
3. Cuvântul este verificat să fie existent în dicționar și să înceapă cu litera aleasa
 - i. În cazul în care cuvântul este valid
 - (a) Jucatorul 3 primește cele 2 litere.
4. Jucatorul 3 va introduce un cuvânt valid care să înceapă cu literele primite.
5. (...)
6. În momentul în care un jucător trimite un cuvânt invalid sau care nu începe cu litera/literele primite, acesta va fi eliminat din sesiune
7. Jocul se continuă până când un singur jucător rămâne în sesiune, acesta fiind declarat câștigător.

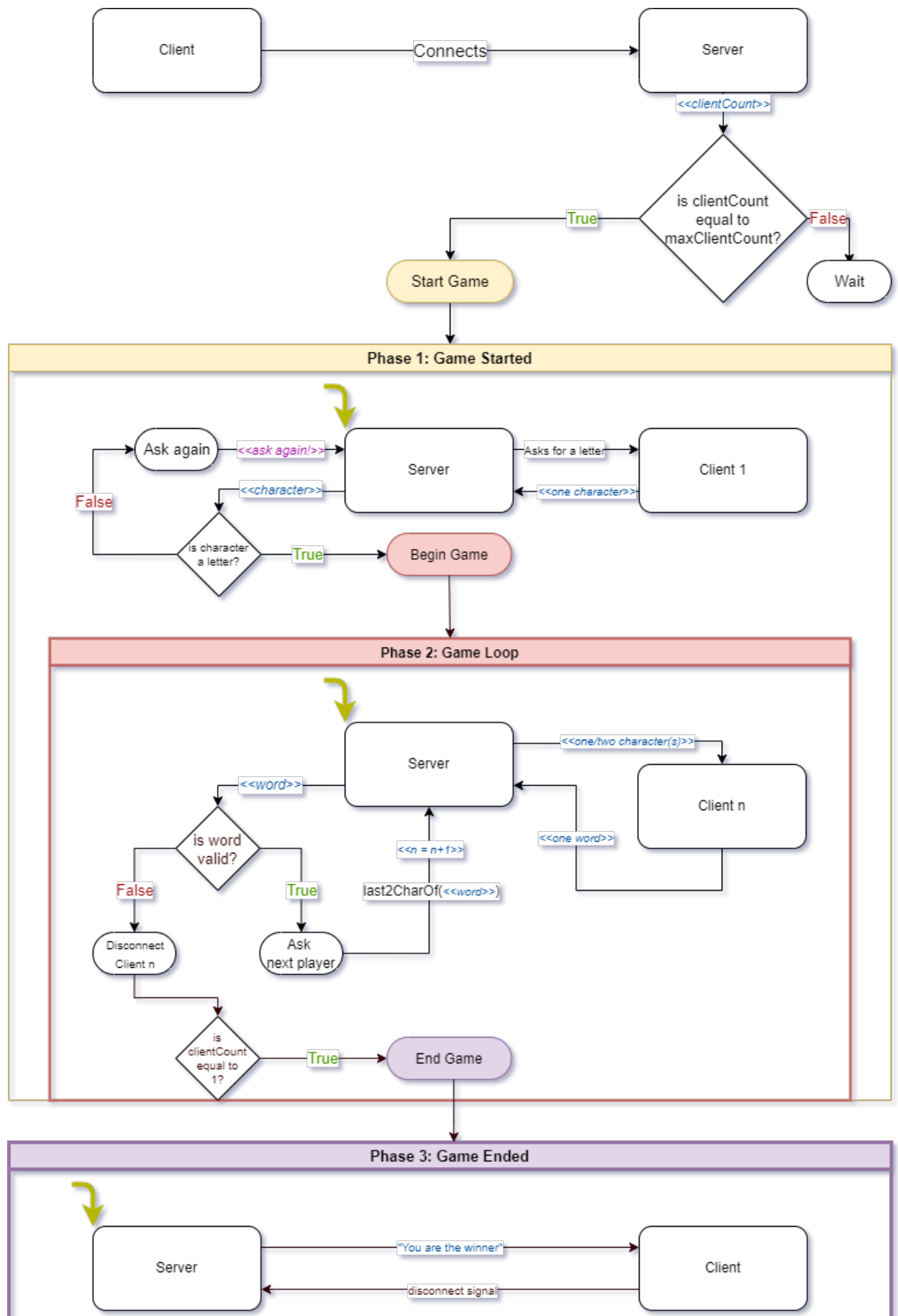
3 Tehnologii utilizate

Pentru acest proiect am folosit Transmission Control Protocol (TCP). TCP este un protocol de transport orientat, cu conexiune, fără pierdere de informații, ce controlează fluxul de date. De asemenea, am folosit varianta TCP concurentă în defavorul celei iterative, deoarece clienții nu ar trebui să fie nevoiți să se aștepte între ei pentru a putea trimite o rezolvare către server. Pentru realizarea interfeței grafice am folosit librăria QT.



TCP server/client Model

4 Arhitectura si detalii de implementare



4.1 Funcțiile grafice

Funcțiile `"drawStartMenu()"`, `"drawGameScene()"`, `"drawLevel()"`, `"drawMove()"` și `"undoLastMove()"` vor utiliza metode din cadrul bibliotecii QT și vor fi responsabile de interfața grafică.

4.1.1 drawLevel()

Această funcție va primi ca argument o matrice de 8x8 reprezentând o situație a unui joc de șah. Pe baza acestei matrici se vor așeza piesele pe tabla desenată precedent de către `drawGameScene()`.

4.1.2 drawMove()

Această funcție va primi ca și argumente o poziție inițială și o poziție finală, prima indică ce piesă se dorește a fi mutată, urmând destinația acestei mutări. Verificarea dacă mutarea este una permisă va fi făcută ulterior de către server.

4.1.3 undoLastMove()

Această funcție va fi apelată în cazul în care serverul decide că mutarea sugerată de către utilizator nu este cea mai optimă și are rolul de a restabili jocul la situația precedentă mutării.

4.2 Username

Jucătorul va trebui să menționeze un username înainte de primirea nivelului, pe baza acestui username, serverul poate extrage din baza de date nivelul curent cât și scorul jucătorului. Dacă username-ul menționat nu este prezent în baza de date, acesta va fi adăugat și scorul lui va fi inițializat cu 0.

4.3 Scorul

Din perspectiva jucătorului, scorul său va fi în permanență afișat pe ecran și va fi incrementat după fiecare nivel completat. Din perspectiva aplicației, clientul va primi scorul de la server pe baza unui username, iar după fiecare nivel completat va trimite serverului noul scor.

4.4 Verificarea mutării

Fiecare mutare făcută de către user va fi trimisă către server pentru a fi verificată. Serverul va verifica mutarea sugerată după două criterii: dacă *este cea mai optimă mutare posibilă* și dacă *este ultima mutare din acest nivel*.

- În cazul în care ambele criterii sunt îndeplinite clientul va cere următorul nivel de la server.
- Dacă doar primul criteriu este îndeplinit, clientul va aștepta o nouă mutare din partea utilizatorului.
- Iar dacă nici măcar primul criteriu nu este îndeplinit, se va apela funcția `undoLastMove()` (4.1.3)

5 Design



Figura 1: Main Menu

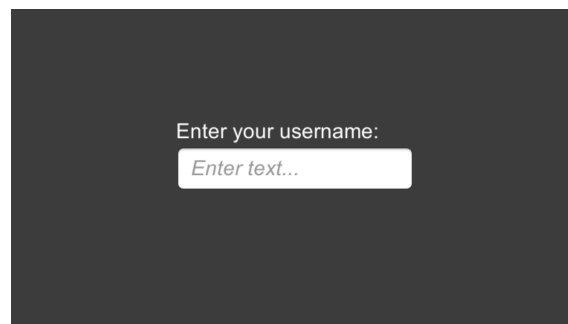


Figura 2: Username input field

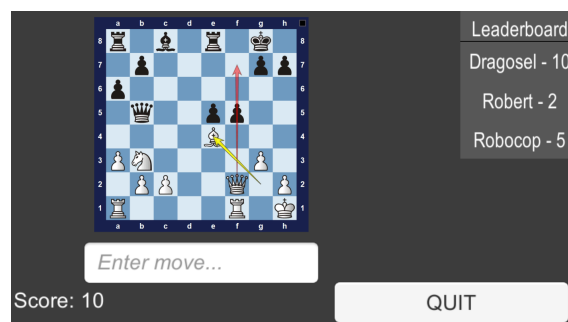


Figura 3: GameView

6 Utilizare

Jucătorul va fi întâmpinat de o interfață grafică și de un buton "Play" (Figure 1). Ulterior va fi nevoit să se logheze folosind un username (Figure 2). După, va apărea primul nivel (Figure 3) și va putea interacționa cu piesele de șah astfel încât să determine cea mai optimă soluție a situației dată. La completarea primului nivel, scorul se va actualiza și următorul nivel va fi afișat.

7 Concluzii

Câteva dintre posibilele îmbunătățiri ale proiectului constau în posibilitatea utilizatorilor să sugereze propriile puzzle-uri, implementarea unui sistem de rating pentru fiecare puzzle și autentificarea prin username și parola.

Bibliografie

Site-ul cursului
SmartDraw (arhitectura aplicației)
Unity (design-ul aplicației)