

Dokumentacja aplikacji

1. Opis projektu

Tematem projektu jest gra sieciowa - szachy zapewniająca jednoczesną rozgrywkę wielu graczom na wielu stołach. Aplikacja zostanie wykonana w architekturze klient - serwer.

2. Zastosowane narzędzia i technologie

Aplikacja korzysta z obowiązkowego API z wykładu do komunikacji sieciowej BSD sockets w języku C. Aplikacja używa też ze standardowej biblioteki C++, bibliotek rozszerzających C++ - boost oraz biblioteki Qt. Graficzny interfejs użytkownika (GUI) jest wykonany w Qt po stronie klienta. Aplikacja przeznaczona jest do działania na systemie operacyjnym Linux, a testowana była na Ubuntu. Automatyczną kompilację aplikacji umożliwia system qmake. Do testów jednostkowych użyta została biblioteka Qt Test.

3. Funkcjonalności

- Funkcjonalności klienta:
 - Przeglądanie lobby aktywnych rozgrywek
 - Tworzenie nowej rozgrywki
 - Dołączenie do już istniejącej rozgrywki
 - Zajęcie miejsca przy stole - po stronie białych lub czarnych
 - Dołączenie do pełnego stołu i obserwacja gry
 - Podczas obserwacji, możliwość dołączenia do gry, gdy ktoś odejdzie od stołu i kontynuacja rozgrywki
 - Zadeklarowanie gotowości do rozpoczęcia rozgrywki
 - Wymiana wiadomości z przeciwnikiem za pomocą okna czatu
 - Opuszczenie gry przed jej końcem i przejście do lobby
 - Wymiana informacji o posunięciach graczy z serwerem
 - Przestrzeganie zasad rozgrywki

- Po zakończeniu gry przejście do lobby
- Funkcjonalności serwera
 - Obsługa wielu gier i klientów w tym samym czasie
 - Synchronizacja postępów rozgrywki między klientami
 - Kontrola przestrzegania zasad rozgrywki
 - Automatyczne kasowanie rozgrywek (stołów), gdy nie będzie w nich żadnego gracza

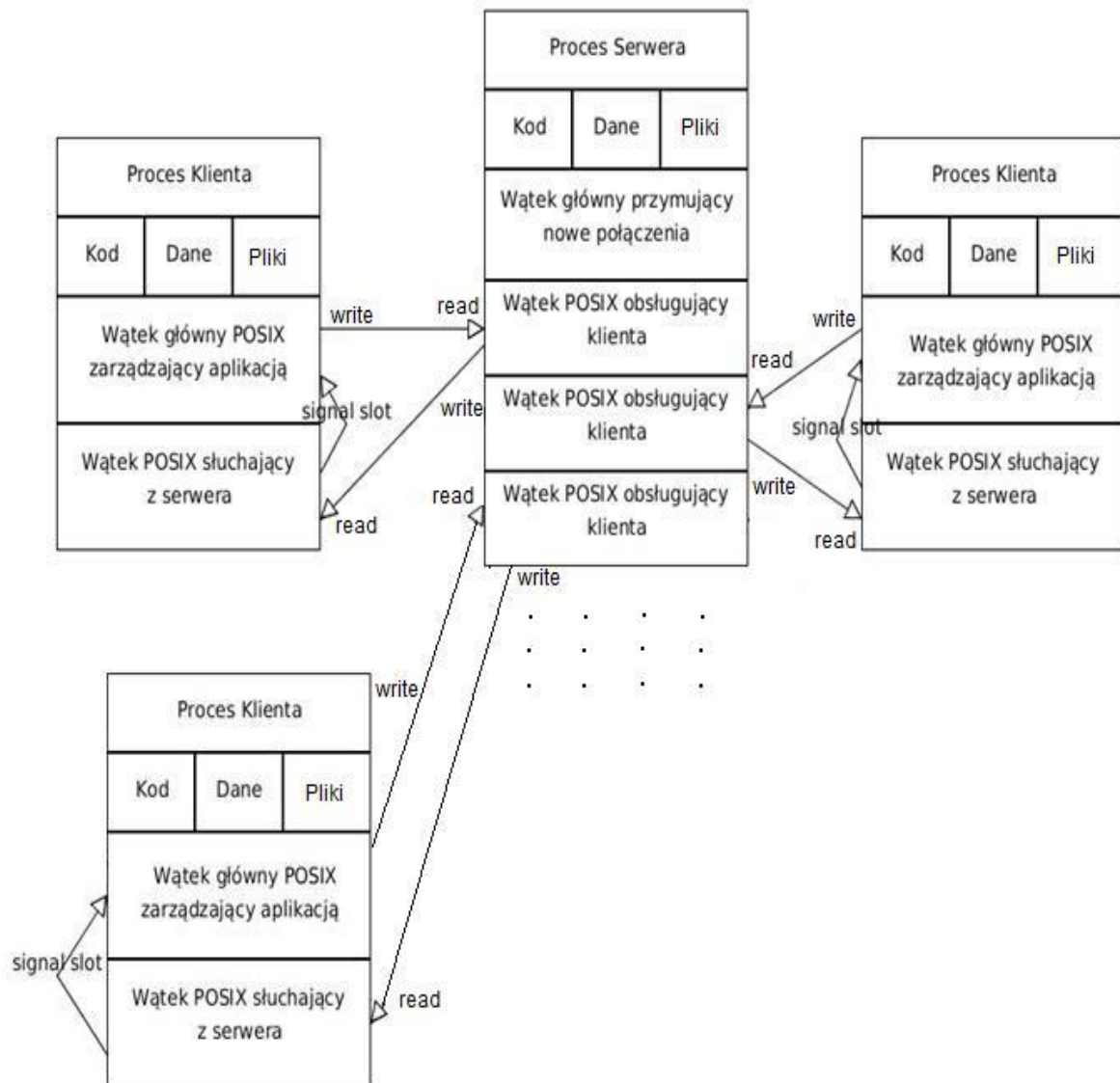
4. Opis architektury

Zaimplementowany został program serwera oraz program klientów łączących się z serwerem.

W serwerze przechowywane są informacje o stołach graczy, liczbie graczy i o rozgrywkach. Programy klientów komunikują się ze sobą za pośrednictwem programu serwera. Programy klientów wyświetlają graficzny interfejs użytkownika, w którym gracze mogą wykonywać różne operacje. Program serwera wykonuje się w konsoli bez graficznego interfejsu i reaguje na komunikaty przychodzące z programów klientów w sposób interaktywny.

5. Opis protokołu komunikacyjnego

Programy klientów będą przysyłać do programu serwera dane o swoim następnym posunięciu, np. parę liczb całkowitych od 1 do 8, kody informacji specjalnych, np. chęć rozpoczęcia rozgrywki (start), rezygnacja z rozgrywki, oraz ewentualnie komunikaty tekstowe w komunikatorze do pozostałych graczy przebywających w danym stole szachowym. Program serwera będzie przysyłać podobne informacje zwrotne do programu klienta należącego do gracza przeciwnego.



Serwer ma główny wątek do łączenia się z nowymi klientami i używa wielu wątków posix thread do obsługi wielu podłączonych klientów, jeden wątek dla każdego klienta. Każdy klient ma główny wątek oraz wątek posix reagujący na komunikaty z serwera.

6. Instrukcja instalacji i uruchamiania aplikacji

Aplikacja została stworzona z użyciem bibliotek Qt, więc trzeba pobrać biblioteki Qt z internetu np. poprzez instalację środowiską Qt stąd <https://www.qt.io/download>

Pierwszym krokiem jest oczywiście pobranie i rozpakowanie archiwum z kodem aplikacji.

Serwer

1. Po pobraniu i rozpakowaniu trzeba przejść w konsoli bash lub cmd do folderu ChessNetworkGameServer, w którym jest zawarty plik ChessNetworkGameServer.pro
2. Zaleca się utworzenie nowego folderu np. build (polecenie *mkdir build* na systemie Linux) i z niego uruchamiamy polecenie *qmake ..* które generuje automatycznie plik Makefile (w windowsie jeśli nie została ustawiona zmienna środowiskowa QMAKE konieczne jest wpisanie całej ścieżki do programu qmake, typu C:\Qt\5.10.0\mingw53_32\bin\qmake.exe)
3. Potem trzeba wywołać polecenie make, które generuje automatycznie działającą aplikację (w windowsie jeśli nie została ustawiona zmienna środowiskowa MAKE konieczne jest wpisanie całej ścieżki do programu make, typu C:\Qt\Tools\mingw530_32\bin\mingw32-make.exe)
4. Następnie można już uruchomić aplikację z pliku ChessNetworkGameServer.exe (w windowsie może być konieczne ustawienie ścieżki do bibliotek Qt w zmiennych środowiskowych lub przekopiowanie bibliotek Qt do folderu w którym znajduje się plik exe, biblioteki Qt znajdują się w ścieżce typu C:\Qt\Tools\QtCreator\bin)
5. Kompilacja aplikacji w systemie Linux powinna przebiegać o wiele sprawniej i prościej niż w systemie Windows. Alternatywnie można też uruchomić aplikację wewnątrz środowiska Qt
6. Aplikacja utworzy połączenie i wyświetli port po którym będzie następować komunikacja.

Klient (command line)

1. Po pobraniu i rozpakowaniu przechodzimy w konsoli bash lub cmd do folderu ChessNetworkGameClient, w którym jest zawarty plik ChessNetworkGameClient.pro
2. Zaleca się stworzenie nowego folderu np. build (polecenie *mkdir build* na systemie Linux) i z niego uruchamiamy polecenie *qmake ..* które generuje automatycznie plik Makefile (w windowsie jeśli nie została ustawiona zmienna środowiskowa QMAKE konieczne jest wpisanie całej ścieżki do programu qmake, typu C:\Qt\5.10.0\mingw53_32\bin\qmake.exe)
3. Wywołujemy polecenie make, które generuje automatycznie działającą aplikację (w windowsie jeśli nie została ustawiona zmienna środowiskowa MAKE konieczne jest wpisanie całej ścieżki do programu make, typu C:\Qt\Tools\mingw530_32\bin\mingw32-make.exe)
4. Następnie uruchamiamy aplikację z pliku ChessNetworkGameClient.exe argumentami wywołania jest adres serwera i port (w windowsie może być konieczne ustawienie ścieżki do bibliotek Qt w zmiennych środowiskowych lub przekopiowanie bibliotek Qt do folderu w którym znajduje się plik exe, biblioteki Qt znajdują się w ścieżce typu C:\Qt\Tools\QtCreator\bin). Kompilacja aplikacji w systemie Linux powinna przebiegać o wiele sprawniej i prościej niż w systemie Windows. Alternatywnie można też uruchomić aplikację wewnątrz środowiska Qt - opis niżej
5. Aplikacja otworzy interfejs graficzny gry.

Klient (Qt Creator)

Po pobraniu i rozpakowaniu przechodzimy do folderu z aplikacją. Otwieramy plik ChessNetworkGameClient.pro w Qt Creatorze. W zakładce Projekty->Uruchamianie w polu "Argumenty linii komend" wpisujemy adres serwera oraz port (oddzielone spacją). Uruchamiamy program - aplikacja otwiera interfejs graficzny gry

7. Graficzny interfejs użytkownika (GUI)



Graficzny interfejs użytkownika składa się z dwóch okien: głównego okna aplikacji reprezentującego lobby ze wszystkimi pokojami (stołami) oraz okna reprezentującego pokój (stół) z rozgrywaną partią szachową.

W głównym oknie znajdują się: nick użytkownika oraz wszystkie stoły z rozgrywkami aktywnymi w danym czasie. Możliwe jest stworzenie nowego stołu lub dołączenie do już istniejącego.

W oknie reprezentującym rozgrywkę widoczna jest szachownica z obrazkami figur. Przesuwanie figur wykonuje się za pomocą funkcjonalności Drag and Drop, tzn. przez przeciąganie figur po planszy. Po kliknięciu na figurę zmienia się kolor odpowiednich pól oznaczających możliwe posunięcia dla wybranej figury. W oknie tym umieszczone są przyciski:

1. Zajmij miejsce
2. Start
3. Opuść miejsce
4. Rezygnacja

Gdy obaj gracze zajmą miejsce i zadeklarują chęć gry (przycisk Start) to grę można uznać za rozpoczętą. Pierwszy ruch należy tradycyjnie do gracza siedzącego po stronie białej. Gdy więcej niż dwóch graczy dołączy do jednego stołu to ci, którzy nie zdążyli zająć miejsca przy stole, mają możliwość obserwacji rozgrywki. Gdy, któryś z grających opuści rozgrywkę, obserwator ma możliwość wejścia na miejsce owego gracza i kontynuacji rozgrywki.

Oprócz tego, w oknie akcji reprezentującej znajduje się pole w którym wyświetlana jest komunikacja pomiędzy graczami przebywającymi w danym pokoju.