

Jakub Borowski
Wojciech Szymski
Hubert Świtalski

03.06.2012

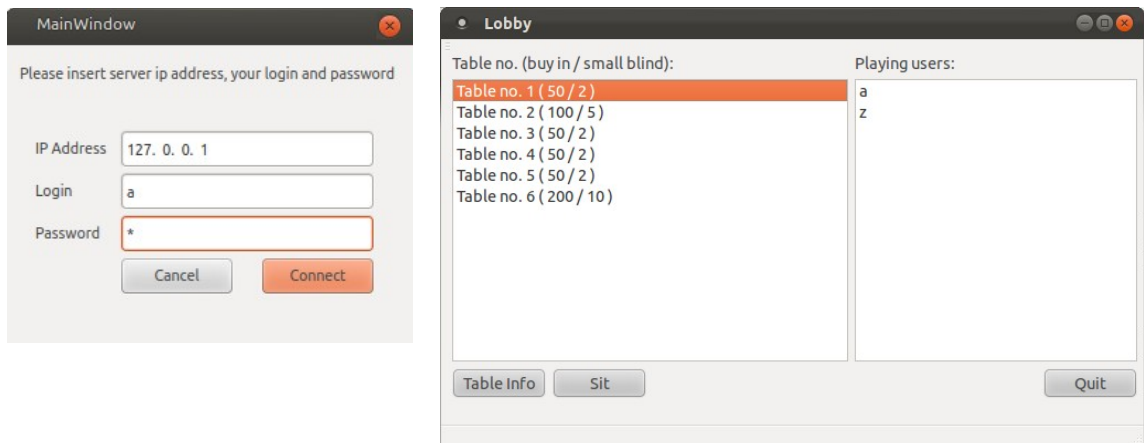
Projekt ZPR
Gra w Pokera
Dokumentacja końcowa

1. Podstawowe informacje o projekcie.

Celem projektu było napisanie sieciowej gry w odmianę pokera Texas Hold'em. Zadanie to zostało w głównej mierze osiągnięte.

Aplikacja została podzielona na dwa niezależne moduły: serwera i klienta. Serwer odpowiada za przebieg rozgrywki, natomiast klient za jego wyświetlanie i przekazywanie decyzji użytkowników.

W uproszczeniu przebieg rozgrywki polega na: połączeniu z serwerem i zalogowaniu się przez klient



wybraniu stołu do gry, a następnie poczekaniu na rozpoczęcie, które następuje gdy na stole pojawi się drugi użytkownik, bądź gdy zakończy się już trwająca partia.



W grze może uczestniczyć do 8 osób przy jednym stole. Zasady gry są klasycznymi zasadami tej odmiany pokera.

2. Zrealizowane funkcjonalności.

- Po stronie klienta:
 - połączenie się z serwerem
 - zalogowanie się na istniejące konto
 - przeglądanie listy aktywnych użytkowników
 - przeglądanie listy aktywnych stołów
 - sprawdzenie szczegółowych informacji o wybranym stole
 - podłączenie się do wybranego stołu
 - odejście od stołu
 - wyrównanie aktualnej stawki
 - podbicie stawki o wybrana kwote
 - spasowanie lub sprawdzenie
 - wyświetlenie informacji o zwycięzcy
- po stronie serwera:
 - komunikacja sieciowa z klientami
 - interpretacja wiadomości od klientów
 - obsługa kont użytkowników
 - zarządzanie lobby (lista użytkowników, list stołów)
 - zarządzanie stołami
 - obsługa gier
 - przetrzymywanie stanu kont w bazie danych
 - możliwość zarządzania częścią rzeczy z poziomu konsoli
 - system uprawnień użytkowników

Po stronie klienta nie została zrealizowana możliwość rejestrowania nowych kont użytkowników. W związku z niewielką ilością osób korzystających z aplikacji uznaliśmy, że tworzenie kont po stronie serwera jest w zupełności wystarczające.

3. Architektura aplikacji.

W naszej aplikacji postawiliśmy na organizację i jakość kodu. Naszym zamiarem było również wykorzystanie powszechnie stosowanych wzorców projektowych i technik programowania, w tym:

MVC (Model View Controller)

Aplikacja składa się z trzech niezależnych warstw: widoku, kontrolera i modelu. Każdy z tych elementów implementuje odpowiedni abstrakcyjny interfejs. Podział zadań jest następujący:

- widok - odpowiada za wyświetlanie okna logowania, lobby z listą stołów i aktywnych użytkowników, a także za wyświetlanie przebiegu

rozgrywki. Odpowiednie dane pobrane od użytkownika umieszcza w modelu, oraz przekazuje do kontrolera akcje użytkownika

- model - przechowuje lokalnie stan rozgrywki, informacje o stole i graczach
- kontroler - w naszej aplikacji kontrolerem jest jednocześnie klasa złużająca za komunikację z serwerem, jak i sam serwer, który przetwarza zapytania graczy

Singleton

Niektóre klasy powinny mieć tylko jedną instancję np. kolejki wiadomości, czy klasa odpowiadająca za połączenie klienta do serwera. Naturalnym rozwiązaniem było zastosowanie singletonu.

Obserwator

Zastosowanie biblioteki QT pozwoliło na użycie w naszym projekcie systemu Sygnałów i Slotów. System ten jest niczym innym jak implementacją wzorca obserwatora.

Narzędzia i biblioteki wykorzystane w projekcie to m.in:

- moduł *QtGui* do tworzenia interfejsu użytkownika
- moduł *QtNetwork* do komunikacji sieciowej
- biblioteki *boost::thread* i *QThread* do tworzenia wątków
- *boost::shared_ptr*
- kontenery STL
- cały projekt powstał w środowisku *QtCreator* i został ujednolicony dla systemów: Windows i Unix
- do tworzenia dokumentacji wykorzystaliśmy narzędzie *Doxygen*
- wykorzystane repozytorium to *Git*

4. Stan aplikacji.

Projekt został zamknięty w pewnej funkcjonalności na potrzeby oceny przez Prowadzącego. Jednak proces tworzenia aplikacji nie został definitywnie zakończony. W trakcie tworzenia pojawiło się wiele dodatkowych pomysłów i funkcjonalności daleko wykraczających poza planowane założenia. Nie wykluczamy, że w przyszłości zajmiemy się dalszym rozwojem gry.

Pomysły na rozwój aplikacji:

- animacja kart
- obliczanie prawdopodobieństwa wystąpienia danego układu, dla konkretnej ręki użytkownika
- obszerniejsze wyświetlanie informacji o grze tj. logi z przebiegu rozgrywki czy też wyświetlanie akcji ostatniego użytkownika
- dodanie efektów dźwiękowych
- poszerzenie informacji o użytkownikach np. e-mail, avatar itp.

- dodanie możliwości interakcji użytkowników poprzez czat lub prywatne wiadomości

W trakcie tworzenia aplikacji spotkaliśmy się z kilkoma trudnościami np. wyścigi wątków klientów po odbiór wiadomości z serwera. Błąd ten powodował, że niektóre komunikaty nie trafiały do docelowego odbiorcy.