



Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Semestrální práce z KIV/UPS

Síťová hra

Student: Tereza Tothová

Osobní číslo: A20B0256P

E-mail: tothova@students.zcu.cz

15. ledna 2021

Obsah

1 Úvod	3
2 Popis zdrojového kódu – server	3
2.1 Klienti a struktura client	3
2.2 Hry a struktura hra	3
2.3 Příchozí komunikace	4
3 Popis zdrojového kódu – klient	4
3.1 Struktura	4
3.3 Grafické rozhraní	4
4 Komunikační protokol	4
5 Překlad programů	8
5.1 Server	8
5.2 Klient	8
6 Uživatelská dokumentace – server	8
6.1 Spuštění serveru	8
7 Uživatelská dokumentace – klient	9
7.1 Login	9
7.2 Lobby	9
7.3 Hra	10
8 Závěr	11

1 Úvod

Zadáním této semestrální práce je implementace síťové hry pro víc hráčů.

Vybranou hrou pro tuto práci je hra jednadvacet. Jednadvacet se hraje s mariášovými kartami a může mít v podstatě libovolný počet hráčů. Na začátku rozdá bankéř každému hráči dvě karty. Hráči si karty prohlédnou a pokud chtějí, žádají ho o další. V okamžiku, kdy už žádný hráč nechce další karty, hráči odkryjí své karty a srovnají své součty bodových hodnot.

Cílem hry je dosáhnout kombinace karet s co nejvyšším součtem bodových hodnot – hráč, který má nejvyšší součet bodových hodnot vyhrává. Zároveň však souhrnná bodová hodnota karet v ruce hráče nesmí překročit číslo 21. Pokud tuto hodnotu překročí, hráč automaticky prohrává.

2 Popis zdrojového kódu – server

Serverová část semestrální práce je realizována v jazyce C.

V souboru server.c se nachází funkce main(), která se spustí jako první a inicializuje server, tj. načte konfiguraci, vytvoří komunikační socket a začne naslouchat na nastaveném portu (default 40000). Funkce také zajišťuje komunikaci s klienty – přijímá spojení, zpracovává příchozí zprávy a odpojuje klienty, kteří hru opustili, nebo byli vyhodnoceni jako nebezpečí pro server z důvodu odesílání nekorektních zpráv.

2.1 Klienti a struktura client

V souboru client.h je definována struktura Client, která drží informace o každém z připojených klientů, zejména jméno, číslo klientského socketu a stav klienta. Je zde také definován list všech klientů Client_list, který je implementován jak spojový seznam a výčet možných stavů klienta. Klient je vždy v jednom z následujících stavů:

- LOGIN
- LOBBY
- WAITING – klient čeká ve hře na protihráče
- IN_GAME
- DISCONNECTED

2.2 Hry a struktura hra

V souboru game.h je definována struktura Game, která představuje jednu hru na serveru, drží si zejména její ID, stav, hráče a balíček karet. Hry se uchovávají v poli her o velikosti, která je určena při startu serveru. Hra je vždy v jednom z následujících stavů:

- NONE – hra ještě nebyla inicializována
- WAITING – hra čeká na hráče
- FULL – hra je plná a může začít
- IN_GAME – hra právě probíhá
- FINISHED – hra došla

Logika hry je obstarávána v souboru `game_manager.c`. Ten využívá balíček karet definovaný v `cards.h`.

2.3 Příchozí komunikace

Zpracování příchozí komunikace zajišťuje soubor `server.c`. Obsahuje smyčku, která přijímá klientské sockety. Jako metoda paralelizace jsou použita vlákna (knihovna `p_threads`) a pro jejich synchronizaci jsou použity mutexy (`p_threads_mutex`). Po připojení nového klienta se vytvoří na serveru nové vlákno pro obsluhu komunikace s klientem. Jakmile dostaneme nová data od klienta, jsou funkcí `parse_in_message()` jsou přeložena na zprávu a dle prefixu zprávy jsou dále zpracovány.

Struktura příchozí zprávy je definována v souboru `message_in.h`. Zpráva obsahuje prefix, argumenty a počet argumentů.

3 Popis zdrojového kódu – klient

Klientská část této semestrální práce je realizována v jazyce Java.

3.1 Struktura

Ve třídě `ClientMain` se nachází funkce `main()`, která se spustí jako první otevře okno pro login.

Třída `ClientTCP` se stará o vytvoření klientského socketu a připojení k serveru.

Ve třídě `MessageInThread` běží smyčka, která přijímá zprávy ze serveru a zpracovává je.

Třídy `Lobby` a `Game` se starají o správné vykreslení oken pro lobby a hru.

Třída `MessageIn` se stará o validaci příchozích zpráv.

Třídy `ClientConstant` a `ClientState` obsahují definované konstanty aplikace a stavy klienta.

3.3 Grafické rozhraní

Pro grafické rozhraní je využita knihovna `JavaFX`. Aplikace využívá tři okna pro tři části hry:

- LOGIN – obrazovka pro zadání uživatelského jména, IP adresy serveru a využívaného portu
- LOBBY – obrazovka se seznamem herních na serveru a s ovládacími prvky pro připojení se k místnosti
- GAME – obrazovka samotné hry

4 Komunikační protokol

Komunikace mezi serverem a klientem se řídí textovým protokolem. Komunikace se dělí na zprávy, které jsou od sebe odděleny středníkem. Argumenty zprávy jsou odděleny mezerou. Jednotlivé zprávy mají formát **<prefix> [params];**. Prefix určuje, z jakého stavu se zpráva posílá. Možné prefixy:

- I – LOGIN
- L – LOBBY
- G – GAME
- E – ERROR
- P – PING

Pokud server obdrží zprávu ve špatném formátu, je klient odpojen od serveru.

Zprávy od klienta pro server:

I <name>	přihlášení
L game <game_id>	klient si vyžádá informace o hře
L enter <game_id>	vstup do hry
L left_the_game	klient úplně opustil hru
G card	klient žádá o další kartu
G no_thanks	klient již nechce další karty
G left_the_game	klient úplně opustil hru
G left_to_the_lobby	klient se vrátil ze hry do lobby
E <error>	chybová zpráva
P pong	klient odpovídá na ping serveru

Zprávy od serveru pro klienta:

I logged_in <name>	úspěšné přihlášení
E error_logging_in	neúspěšné přihlášení
L number_of_games	počet otevřených her
L info <game_id><state><n_players>	informace hře
L entered <game_id><n_players>	úspěšný vstup do hry
L cant_enter	neúspěšný vstup do hry
G game_started	hra začala
G card <name><value><pattern >	poslaná karta
G player <name>	jméno protihráče
G win	výhra
G lose	prohra
G player_left <name>	hráč odešel ze hry
G player_disconnected <name>	hráč neodpovídá na ping
G player_reconnected <name>	hráč se připojil zpět do hry
E <error>	chybová zpráva
P ping	server posílá ping klientovi

KLIENT:

I <name> přihlášení

L game <game_id> klient žádá info o hře

L enter <game_id> klient žádá vstup do hry

L left_the_game klient úplně opustil hru

Dostatek hráčů ve hře

G card klient žádá o další kartu

G no_thanks klient nechce další karty

Nikdo již nechce karty

G left_the_game klient úplně opustil hru

G left_to_the_lobby klient odešel do lobby

Hráč se odpojí

Hráč se připojí zpět

SERVER:

I logged_in <name> úspěšné přihlášení

L number_of_games počet otevřených her

E error_logging_in neúspěšné přihlášení

L info <game_id> <state> <n_players>

L entered <game_id> <n_players> úspěch

L cant_enter neúspěch vstup

server ukončí spojení a smaže klienta

G game_started hra začala

G player_name jméno hráče

G card <name><value><pattern> karta

G full_hand_of_cards neúspěch karta

server ukončí hru, pokud nikdo nechce karty

G win

G lose

server ukončí hru, pokud není dost hráčů

G player_left <name> pošle ostatním

server přesune hráče do lobby

G player_left <name> pošle ostatním

G player_disconnected <name> ostatním

G player_reconnected <name> ostatním

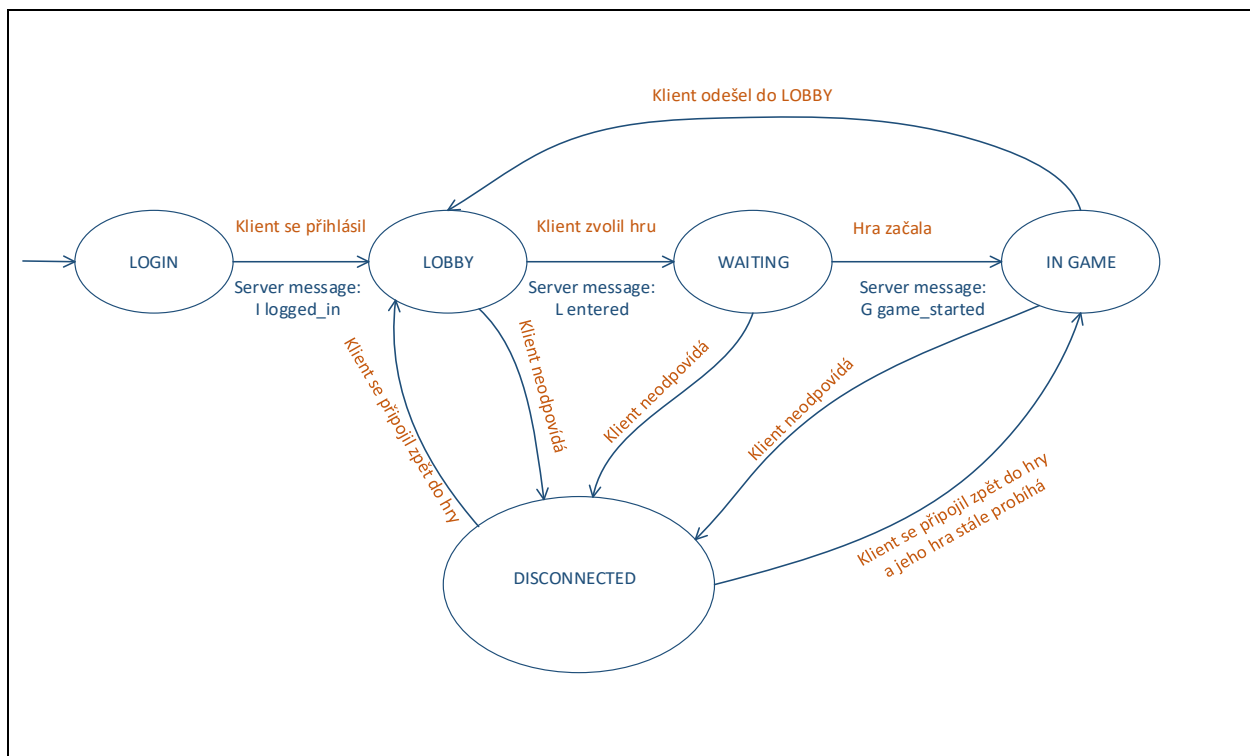


Figure 1 Stavový diagram klienta

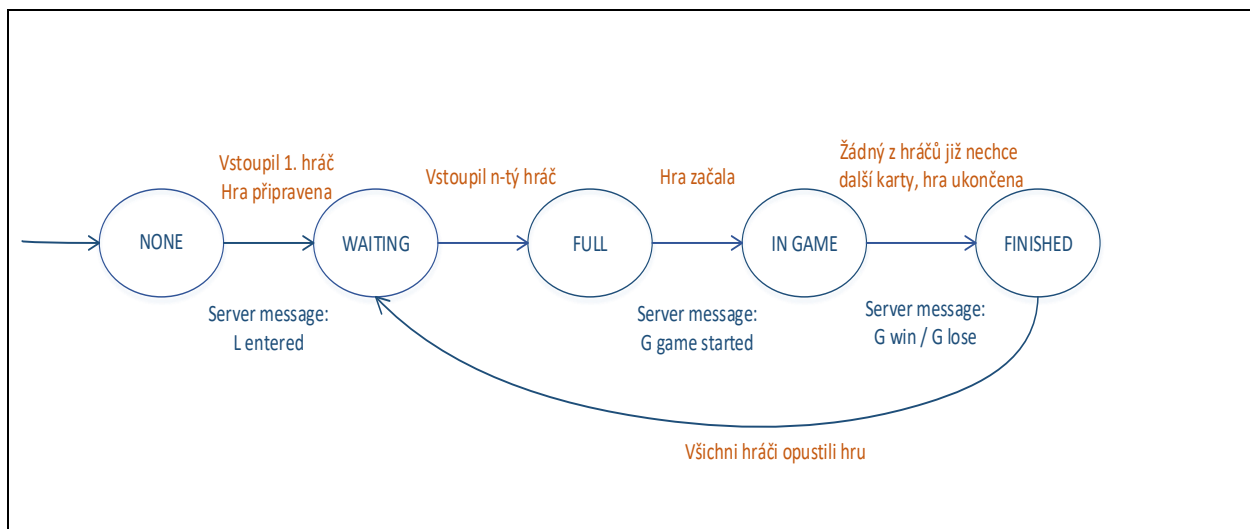


Figure 2 Stavový diagram hry

5 Překlad programů

5.1 Server

Pro úspěšný překlad serverového programu je nutné mít nainstalován překladač jazyka C , sestavovací systém CMake verze alespoň 3.13.

Překlad se provádí následujícími příkazy v adresáři s projektem“

```
mkdir build
```

```
cd build
```

```
cmake ..
```

```
make
```

Po úspěšném překladu se v adresáři build vytvoří spustitelný server Sp1.

5.2 Klient

Pro úspěšný build a spuštění klienta je potřeba mít nainstalován Apache Maven. Ve složce se souborem pom.xml a složkou zdrojových kódů src je klient přeložen pomocí následujících příkazů.

```
mvn package
```

```
cd target
```

```
java -jar ...jar
```

6 Uživatelská dokumentace – server

6.1 Spuštění serveru

Server je možno spustit buď bez argumentů nebo s jedním až pěti argumenty v libovolném pořadí.

-address || -a [IPv4] - IP adresa (default je jakákoli)

-port || -p [number] - port (1024-49151, default 40000)

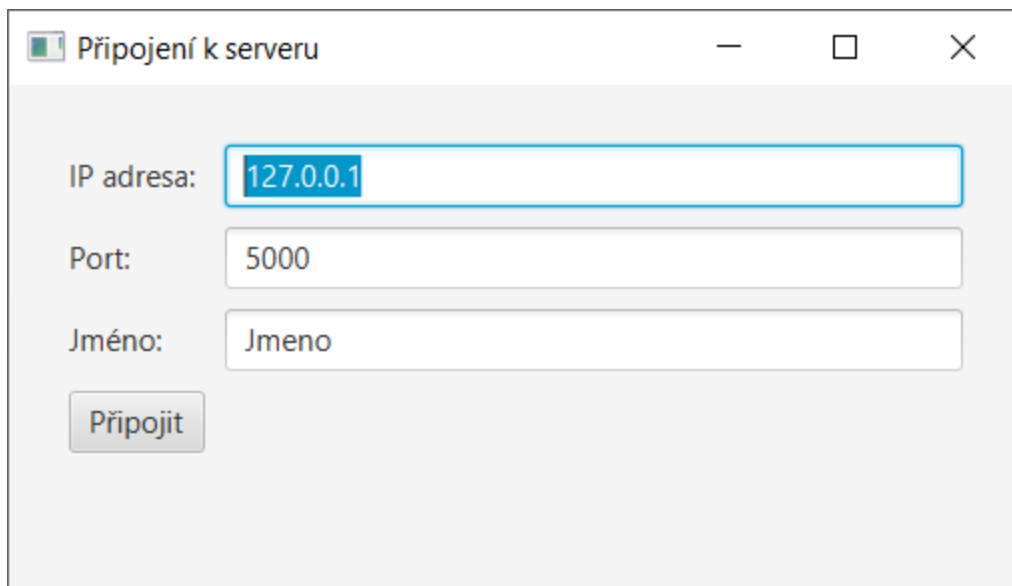
-games || -g [number] - počet her (1-10, default 5);

-players || -pl [number] - počet hráčů ve hře (2-5, default 3)

Příklad: ./Sp1 -a 127.0.0.1 -p 10000 -g 3 -pl 2

7 Uživatelská dokumentace – klient

7.1 Login



Připojení k serveru

IP adresa: 127.0.0.1

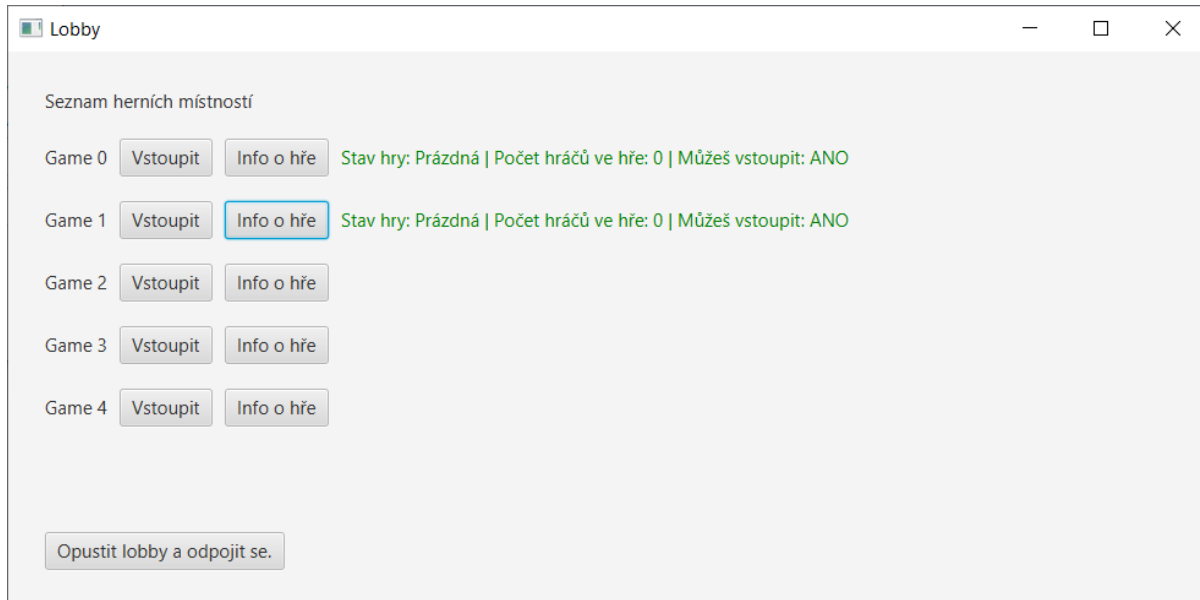
Port: 5000

Jméno: Jmeno

Připojit

Obrázek 1 Login okno

7.2 Lobby



Lobby

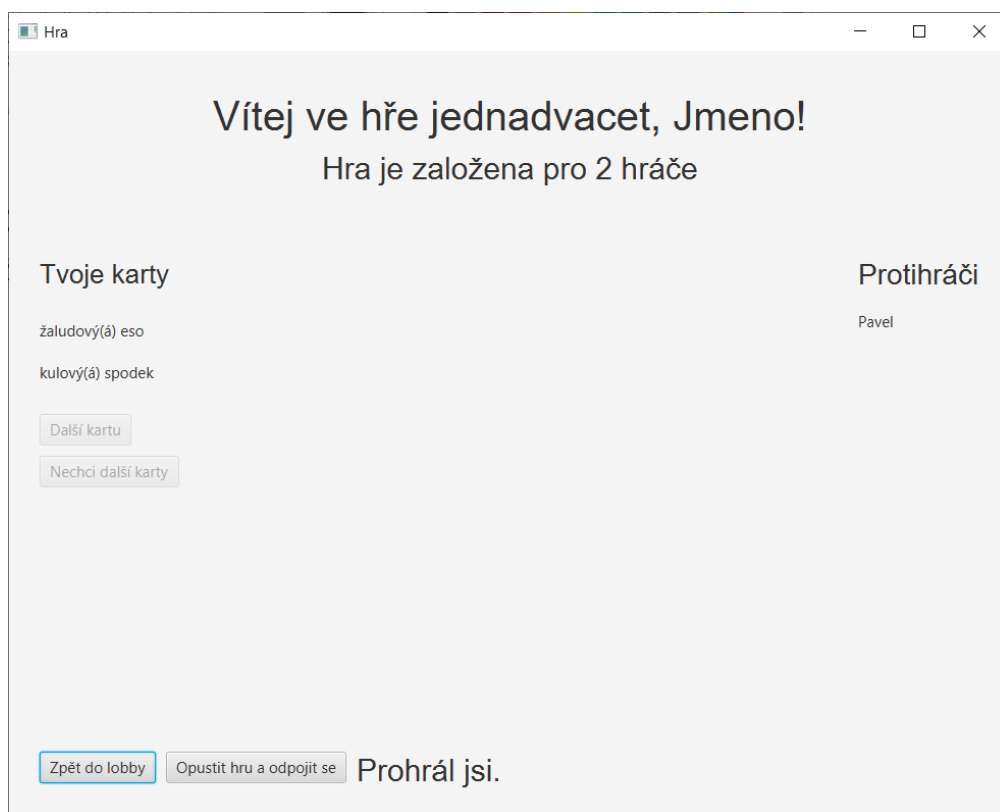
Seznam herních místností

Game 0	Vstoupit	Info o hře	Stav hry: Prázdná Počet hráčů ve hře: 0 Můžeš vstoupit: ANO
Game 1	Vstoupit	Info o hře	Stav hry: Prázdná Počet hráčů ve hře: 0 Můžeš vstoupit: ANO
Game 2	Vstoupit	Info o hře	
Game 3	Vstoupit	Info o hře	
Game 4	Vstoupit	Info o hře	

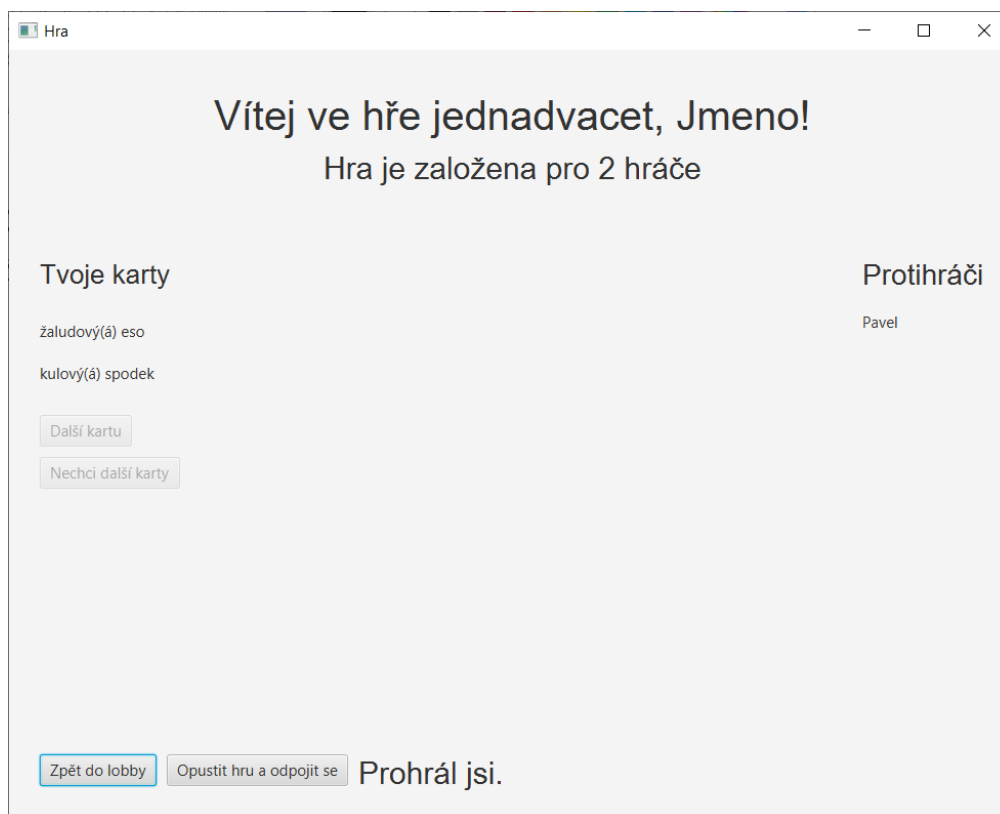
Opustit lobby a odpojit se.

Obrázek 2 Lobby okno

7.3 Hra



Obrázek 3 Game okno



Obrázek 4 Game okno

8 Závěr

V rámci semestrální práce vznikly dva programy, které jsou spolu schopny vzájemně komunikovat pomocí textového protokolu nad protokolem TCP.

Počet hráčů ve hře je omezen pouze pravidly hry jednadvacet, je tedy možné hrát hru ve třech hráčích. Ukončení jednoho klienta při probíhající hře nechá ostatní hráče dohrát, pokud jsou aspoň 2 a má tak hra smysl. Obě aplikace ošetřují nevalidní zprávy a odpojí protistranu při chybě.

Možným rozšířením by bylo například propracovanější grafické rozhraní, případně registrace hráčů, díky které by se mohli hráči vracet do hry i po výpadku serveru.