

Západočeská univerzita v Plzni  
Fakulta aplikovaných věd  
Katedra informatiky a výpočetní techniky

**Bakalářská práce**

**Hra "Mlýn" na síti**

Plzeň, 2008

Josef Bacík

List se zadáním

## Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 26.6.2008

.....

## **Abstrakt**

The aim of the work was to make a network application enabling to play various board games on the net. Within the frame of this work it was necessary to learn the rules of “Merels” game, to explore the modifications of the game and to choose suitable rules for its realisation as well as to analyse the possibilities of this game in the network environment, to choose the best protocol form and the communication system between the players. The next aim was to design the interface and application display based on requirements coming out of expansion of the hardware and the software. Display the game in tree-dimensional vision and to adapt the rules in a way enabling to play the game. The last part was to verify the solution and to accomplish the user’s and programmer’s documentation.

## Obsah:

1	Úvod .....	7
2	Hra „Mlýn“ .....	8
2.1	Historie hry „Mlýn“ .....	8
2.2	Popis hry „Mlýn“ .....	8
2.3	Pravidla a cíle hry .....	9
2.4	Alternativní pravidla .....	9
2.5	Alternativní mapy .....	10
2.6	Pravidla vybraná pro realizaci programu .....	10
3	Možnosti hry v prostředí počítačových sítí .....	12
3.1	Síťový model .....	12
3.2	Protokol TCP .....	12
3.3	Protokol UDP .....	13
3.4	Požadavky na protokol .....	13
3.5	Výběr vhodného protokolu .....	14
3.6	Implementace spojení .....	14
3.7	Neveřejná IP adresa .....	14
3.8	Možnosti komunikace mezi hráči .....	15
3.9	Výběr způsobu komunikace mezi hráči .....	16
4	Návrh rozhraní pro zobrazení a ovládání hry .....	17
4.1	Požadavky na okno herní aplikace .....	17
4.2	Zpracování okna programu .....	17
4.3	Zobrazení hry .....	17
4.4	Ovládání hry .....	18
4.5	Zobrazení herního serveru .....	19
5	Další funkce herního klienta a možná rozšíření .....	20
5.1	Profily a nastavení sítě .....	20
5.2	Zvuky .....	21
6	Možné programovací jazyky pro realizaci programů .....	22
6.1	JAVA .....	22
6.2	C# .....	22
7	Realizace síťového rozhraní serveru pro hru „Mlýn“ .....	23
7.1	Herní Server .....	23
7.2	Požadavky na herní server .....	23
7.3	Jak se chová server při připojení hráče .....	23
7.4	Možnosti serveru .....	25
8	Realizace síťového rozhraní pro klientskou aplikaci .....	26
8.1	Windows klient .....	26
9	Realizace uživatelského rozhraní a grafického výstupu .....	27
9.1	Prvky Windows Forms .....	27
9.2	Prostorové zobrazení pomocí DirectX .....	27
9.3	Návody her .....	28
9.4	Klient pro PDA .....	28
10	Sumarizace požadavků na HW a SW počítače .....	29
10.1	Server .....	29

10.2	Windows klient.....	29
10.3	PDA klient.....	29
11	Získané poznatky a ověření funkčnosti.....	30
11.1	Získané poznatky.....	30
11.2	Ověření funkčnosti.....	30
12	Závěr.....	31
	Literatura.....	32
	Příloha A – Uživatelská dokumentace ovládání.....	33
	Ovládání serveru.....	33
	Ovládání Windows klienta.....	34
	PDA klient.....	37
	Příloha B – popis instalace.....	39
	Server 39	
	Klient WIN.....	41
	Klient PDA.....	42
	Příloha C – Programová dokumentace.....	43
	Herní server.....	43
	Klient Windows.....	45
	PDA klient.....	46

# 1 Úvod

Trendem současné doby zejména u mladé generace je obliba počítačových her. Deskové hry v klasické podobě bývají často opomíjeny. Svou prací se snažím o zpopularizování těchto her, konkrétně hry mlýn. Hra rozvíjí logické myšlení, učí řešit problémy a předvídat situace. Původní pravidla jsem upravil tak, aby byly využity možnosti, které poskytuje hra na počítači např. další tahy v prostorovém modelu mlýnu. Hra po internetu má některé přednosti, např. hráči se nemusí fyzicky sejit, přezdívky umožňují jistou anonymitu.

Hru jsem navrhl jako síťovou aplikaci využívající herní server obstarávající komunikaci mezi hráči tak, aby i hráči, kteří se připojí z neveřejné IP adresy<sup>1</sup> mohli zakládat hry a plnohodnotně využívat všech možností hry. Pro klientský program byla aplikace realizována pomocí grafických knihoven do trojrozměrného zobrazení.

Serverová aplikace může být díky své obecnosti využívána pro přenos herních dat libovolné deskové hry, přičemž jedna serverová aplikace může být využívána pro přenos dat více typů her najednou. Stejně tak i klientská část je z velké části využitelná pro implementaci dalších deskových her.

---

<sup>1</sup> Neveřejná IP adresa znamená, že počítač s touto adresou není viditelný z internetu. Jedná se IP adresu vnitřní sítě. Z tohoto důvodu většinou nemůže jiná aplikace s touto navázat spojení přes internet. Podrobněji vysvětleno v kapitole 3.6.

## 2 Hra „Mlým“

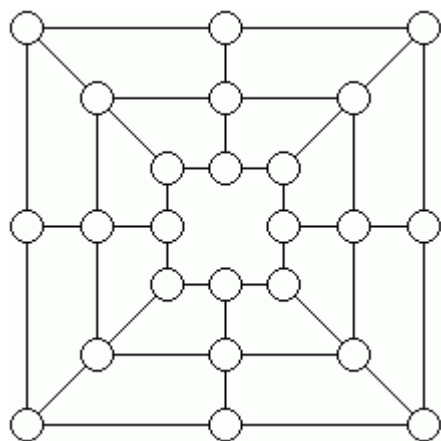
### 2.1 Historie hry „Mlým“

Hra Mlým je strategická desková hra pro dva hráče, jejíž nejstarší herní mapa pochází ze starověkého Egypta a období vzniku první mapy se odhaduje na rok 1400 před naším letopočtem. Předpokládá se, že tuto hru znali i staří Římané, lze se tak domnívat na základě nálezů herních map. Bohužel ani zde se nepodařilo určit přesné stáří. Další mapy byly také nalezeny v Anglii v období středověku. Hry se často rozcházejí v názvu a způsobu zpracování herního plánu, například mapy v Anglii měly místo pro kameny vyznačeny vyrytými dírami – odtud název „nine holes“. Jiné názvy podle odhadů vycházejí ze slova merellus, které označuje šachového pěšce nebo hrací kámen. Dalšími anglickými názvy pro tuto hru jsou: Nine Men's Morris, Nine Man Morris, Mill, Mills a Merels.[2]

Rozšířenost hry tak způsobila nejen vznik různých názvů, ale vznikly také odlišnosti v herních pravidlech a řady modifikací základních map. Proto je třeba u každé hry specifikovat pravidla.

### 2.2 Popis hry „Mlým“

Pro hru mlým je zapotřebí herní mapa a 18 hracích kamenů (např. 9 bílých a 9 černých). Mapy jsou různé podle konkrétní země (další např. viz. obr. 1, 2, 3). V Čechách je rozšířena především mapa tvořena třemi čtverci, jež mají společný střed a jsou protnuty čtyřmi úsečkami procházející středy stran a čtyřmi úsečkami spojujícími vrcholy čtverců, viz obr. 1.



Obrázek 1: Herní mapa mlýnu



## **2.3 Pravidla a cíle hry**

Hra se hraje s osmnácti kameny – devět bílých a devět černých. Partii zahajuje hráč s bílými kameny. Položí jeden ze svých kamenů na kterýkoliv bod, kde se sbíhají nebo protínají různoběžné linie. Těchto bodů je na klasické herní desce (obr. 1) celkem čtyřadvacet. Po tahu hráče s bílým kamenem je na tahu protihráč s černým kamenem. Také on položí svůj kámen na kterýkoliv volný bod. Dál se oba v kladení kamenů pravidelně střídají a snaží se vytvořit „mlýn“. Za mlýn považujeme tři kameny stejné barvy ležící na jedné čáře. Hráč, který položil ke dvěma svým kamenům třetí a dokončil tím mlýn, smí odebrat soupeři jeden kámen. Pravidla však zakazují odebrat kámen z již dokončeného mlýnu. Když nás právě dokončený mlýn opravňuje k odebrání protivníkova kamene, vezmeme ten kámen, který by mu mohl v některém z nejbližších tahů dokončit mlýn. Pokud má protihráč všechny kameny v mlýnu, odebere kterýkoli kámen z mlýnu.

V případě, že hráč který hru začínal je na tahu a již nemá žádné kameny, které by mohl rozmístit na hrací plochu, začíná fáze hry, ve které hráči posunují střídavě své hrací kameny po liniích na hrací ploše. Při jednom tahu smí každý přemístit jen jediný kámen na nejbližší bod, který musí být prázdný.

Všechny operace, které s kameny provádíme, směřují k jednomu ze dvou cílů – jednak se snažíme vytvořit vlastní mlýn, ale kromě toho chceme zabránit soupeři v dokončení jeho mlýnu. Správná strategie počítá s oběma úkoly.

Pravidla v různých zemích se rozcházejí i v závěru hry ve fázi, kdy má jeden z protihráčů pouze tři kameny. Zde pravidla umožňují tomuto hráči skákat po čarách přes volné pole i přes soupeřův kámen, aby zabránil protivníkovi uzavřít mlýn nebo aby sestavil svůj vlastní mlýn. Některá pravidla umožňují umístit svůj kámen dokonce na libovolnou volnou pozici na herní mapě. Naopak jiná pravidla tento stav vůbec neřeší a platí při něm pravidla jako při hře s více než třemi kameny.

Kdo ztratil jeden z posledních tří kamenů, prohrál. U her, kde není ošetřen stav, kdy má hráč pouze tři kameny, je možné hru ukončit v okamžiku, kdy již pro hráče na tahu neexistuje tah podle pravidel.

## **2.4 Alternativní pravidla**

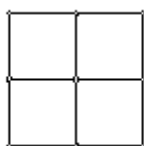
Hraje se s různým počtem kamenů, např. 6, 10, 14.

Je přidáno pravidlo (při mezinárodních turnajích), podle kterého nesmí být jeden mlýn uzavřen dvakrát za sebou ve stejné sestavě kamenů. Před novým uzavřením v něm musíme

změnit pořadí kamenů nebo jeden z kamenů nahradit jiným, který při minulém uzavření ve mlýnu nebyl.

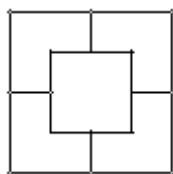
## 2.5 Alternativní mapy

Three Men's Morris je název pro herní mapu, která je tvořena jedním čtvercem a dvěma úsečkami spojující středy protilehlých stran. V této variantě je možné místo posunů kamenů skákat. Stejná pravidla lze aplikovat i na mapu, která je oproti této rozšířena o úhlopříčky, viz. obr 2.



Obrázek 2: Herní mapa hry Three Men's Morris

Six Men's Morris je mapa se dvěma čtverci propojenými čtyřmi úsečkami, přičemž každá úsečka vychází ze středu hrany menšího čtverce směrem ven a končí na středu hrany většího čtverce. Oproti mapě Three Men's Morris je na této mapě není dovoleno skákat. Viz obr. 3.



Obrázek 3: Herní mapa hry Six Men's Morris

Twelve Men's Morris - v této hře jsou navíc úhlopříčky spojující rohové body a hráči hrají každý s dvanácti kameny, což může způsobit, že už při kladení kamenů dojde k obsazení celého herního pole a hra končí remízou.

Pravidla pro hru „Mlýn“ a mapy jsou z [2]

## 2.6 Pravidla vybraná pro realizaci programu

Z množství pravidel je třeba specifikovat pravidla, kterými se bude řídit program při zpracování uživatelského vstupu a rozesílání tahů.

Počet hracích kamenů bude stanoven na 18 (9 bílých a 9 červených). Hru zahájí hráč s bílými kameny. Vezme jeden svůj volný kámen a položí ho na libovolné místo na mapě. Následuje tah protihráče, který rovněž pokládá svůj kámen na kterékoli volné místo na hrací

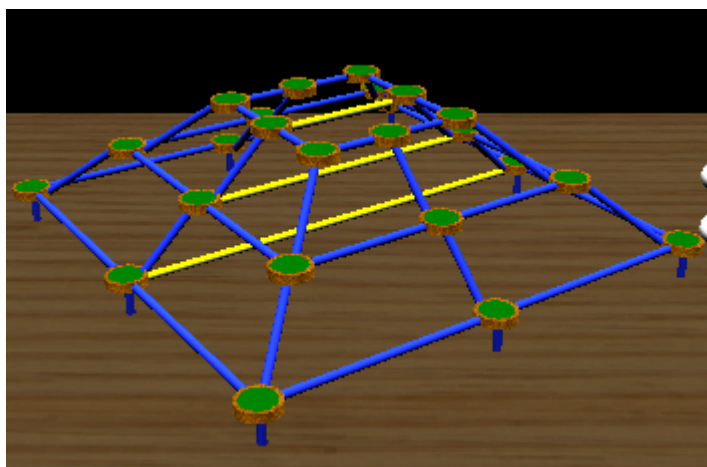
ploše. Takto se v tahu oba střídají do doby, než jeden z nich udělá „mlýn“. „Mlýnem“ se rozumí trojice hráčových kamenů uspořádaná na jedné linii. Hráč, který takto vytvoří „mlýn“, odebírá protihráčův kámen. Odebírá jen ten kámen, který není v „mlýnu“ s jinými kameny. Pouze v případě, že by měl protihráč všechny kameny v „mlýnu“, může odebrat libovolný kámen. Když každý z hráčů položí svůj devátý kámen, začínají se kameny posouvat. Posunutí kamenu lze realizovat pouze o jednu pozici za předpokladu, že je cílové místo přesunu kamene volné. Pokud hráč vytvoří svými kameny „mlýn“, je oprávněn odebrat soupeři kámen. Odebrání v průběhu hry se řídí stejnými pravidly jako při pokládání kamenů. Pokud hra dospěje do fáze, že má jeden z hráčů pouze tři kameny, je tento hráč oprávněn k provádění přesunů přes celou linii a to i přes soupeřovy kameny. Ten z hráčů, který bude mít jako první pouze dva kameny, prohrál.

Ve hře nejsou aplikovaná mezinárodní pravidla, hráč tak může vytvořit opakovaně mlýn stejným kamenem (například tak, že z „mlýnu“ v jednom tahu odebere jeden z kamenů a v následujícím tahu tento kámen zase vrátí na původní pozici).

Herní mapa bude použita jako mapa na obr. 1.

Vzhledem k použité herní mapě a zvolenému počtu herních kamenů, by nemělo dojít ke stavu, že by měl jeden hráč zablokována všechna pole a hra tak končí až porážkou jednoho z hráčů.

V programu je také herní mapa mlýnu, umožňující zobrazení do prostoru. Tato mapa má podobný tvar jako mapa na obr. 1, ale každý z vnitřních čtverců je posunut výše, čímž vzniká jakoby pyramida viz obr. 4. Navíc jsou přidány tři nové linie (žlutě vyznačené na obr. 4) umožňující táhnout vnitřkem mlýnu a využít tak třetí rozměr. Pravidla pro prostorový mlýn jsou totožná s výše uvedenými pravidly v této podkapitole. „Mlýn“ je možné vytvořit opět pouze trojicí kamenů jednoho hráče na jedné linii.



Obrázek 4: Zobrazení prostorového modelu

## **3 Možnosti hry v prostředí počítačových sítí**

### **3.1 Sít'ový model**

Sít'ová komunikace se dnes v počítačových sítích realizuje především pomocí sít'ových protokolů TCP/IP, které jsou nejpoužívanější sít'ovou technologií. Protokoly TCP/IP fungují nad libovolnou přenosovou technologií (až na výjimky se nezabývají fyzickou a linkovou vrstvou) a většina aplikací je implementována na tyto protokoly. Architektura TCP/IP je sestavená ze čtyř vrstev. Vrstva sít'ového rozhraní, sít'ová vrstva, transportní vrstva a aplikační vrstva.

Vrstva sít'ového rozhraní slouží pro přístup k fyzickému přenosovému médiu.

Sít'ová vrstva obsluhuje adresování, směrování a předávání datagramů pomocí protokolů např.: IP (vysílá datagramy a poskytuje vyšším vrstvám sít'ovou službu bez spojení – je nespojovaný a nespolehlivý), ARP (k získání ethernetové adresy sousedního stroje), ICMP (přenáší řídicí hlášení o chybách v přenosu).

Transportní vrstva je již implementována v koncových zařízeních a zajišťuje potřebné přizpůsobení sítě potřebám aplikace. Zajišťuje komunikaci mezi koncovými účastníky. Umožňuje měnit charakter služeb a přenosu mezi spolehlivým a spojovým (protokoly TCP) a mezi nespolehlivým a nespojovaným (protokoly UDP).

Poslední vrstvou je aplikační vrstva, sem patří programy využívané ke konkrétním službám, například: HTTP (internetový protokol pro přenos dokumentů), FTP (přenos souborů), DNS (převod doménových jmen a IP adres). Protokoly aplikační vrstvy využívají vždy alespoň jeden protokol z transportní vrstvy (tj. TCP nebo UDP). Pro rozlišení aplikačních protokolů se pak využívá port, což je domluvený kanál komunikace.

### **3.2 Protokol TCP**

Protokol TCP odpovídá transportní vrstvě. Tento protokol je spolehlivý a emuluje spojovaný způsob fungování.

Spojovaný způsob komunikace znamená, že strany stanic, které spolu chtějí komunikovat, nejprve navážou spojení. Tím je vytyčena trasa přenosu, kterou budou obě stanice používat ke komunikaci. Dále se ověřuje, jestli spolu tyto stanice opravdu chtějí komunikovat, popřípadě se určí parametry komunikace. Vlastní komunikace pak probíhá po trase, která byla nalezena při vytvoření spojení. Na konci komunikace je třeba spojení ukončit, čímž se zruší vytyčená trasa a uvolní přidělené zdroje.

Výhoda spojované komunikace je tedy v tom, že připojení se drží po celou dobu přenosu a při doručování dat je zachováno pořadí, ve kterém byla data odeslána.

Spolehlivý přenos znamená, že služba starající se o přenos dat kontroluje, jestli byla data skutečně doručena. Pokud zjistí, že došlo k poškození nebo ztrátě dat, začne se starat o nápravu. Obvykle si služba vyžádá opětovné přenesení těchto dat.

Při zajišťování spolehlivého přenosu se ale také zvyšuje režie, která se negativně projeví potřebou větší přenosové kapacity a delším časem zpracování, protože přenos musí zjišťovat, jestli jsou data v pořádku a v případě chyby čekat na opětovné přijetí nových dat. Výhodou je jistota, že data budou v pořádku doručena.

### **3.3 Protokol UDP**

Protokol UDP také odpovídá transportní vrstvě. Hlavní rozdíly při porovnání s protokolem TCP spočívají v tom, že UDP je protokol nespolehlivý s nespojovanou komunikací.

Nespojovaná komunikace narozdíl od spojované neprochází žádnými stavy. Nemá stavy navazování a ukončování spojení. Aplikační data jsou obsažena již v prvním segmentu. Bloky přenášených dat musí nést v hlavičce plnou adresu příjemce.

Důležitou skutečností je fakt, že data přenášená při nespojované komunikaci nemusí zachovávat pořadí, v jakém byla odvysílána.

Nespolehlivý protokol znamená, že ačkoli se přenosová služba snaží o korektní přenos a nepoškození či ztrátu dat, mohou tyto chyby přenosu vzniknout. Služba ale při zjištění chyby na tuto skutečnost nereaguje žádnou snahou o nápravu, chybná data zahodí a pokračuje v přenosu. Předpokládá, že pokud bude potřeba nastalou ztrátu dat nějak řešit, postará se o to vyšší vrstva.

Výhodou nespolehlivého přenosu je fakt, že s ním je spojena malá režie, která je v TCP potřeba ke kontrole či opravě poškozených dat, tím se přenos také stává rychlejším oproti TCP.

V kapitolách 3.1, 3.2 a 3.3 je čerpáno z [1].

### **3.4 Požadavky na protokol**

Jelikož bude síť využívána pro potřeby hry mlýn, popřípadě jiné deskové hry, je třeba vybrat vhodný protokol na základě potřeb těchto her.

V deskových hrách se budou realizovat především přenosy, u kterých bude kladen důraz na to, aby data dorazila v pořádku. Například je nutné, aby řetězec obsahující data

o stavu hry dorazil celý, přičemž není kladen takový nárok na rychlost přenosu jako by mohl být při přenosu například audio nebo video signálu. Vzhledem k tomu, že bude potřeba přenášet různá data, bude vhodné každý řetězec opatřit identifikátorem, díky kterému příjemce jednoznačně rozpozná, jaká data jsou přenášena.

### **3.5 Výběr vhodného protokolu**

Na základě požadavků na protokol byl zvolen protokol TCP. Ten byl vybrán pro vlastnosti, díky kterým se dokáže postarat o vzniklé chyby během přenosu. Jeho další předností je spojovanost, které bude využito pro zjištění přerušování spojení, a tak bude jednoduché toto přerušování detekovat a oznámit uživateli textovou hláškou. Při přerušování pak bude celé spojení ukončeno a uživateli bude nabídnuta možnost připojení opět navázat. Pokud by došlo k ukončení připojení během hry, nebylo by již možné ve hře pokračovat. Výhodou spojovaného připojení je také zachování pořadí odeslaných zpráv. Díky tomu se bude moci v případech, kdy bude jasně dáno, jaká data se mají posílat, odesílat tato data bez určeného identifikátoru. To lze využít například při úvodní komunikaci.

### **3.6 Implementace spojení**

Vybraný protokol TCP umožňuje spojově orientovanou komunikaci. Ta je dobře realizovatelná, neboť pracuje s proudy dat (stream), velmi podobně jako při práci se soubory. Díky této abstrakci se program nemusí zabývat tím, jak je komunikace řešena. Stačí mu, pokud získá příslušný proud. Pro sestavení spojení se použije socket (koncový bod obousměrného komunikačního toku), který obsahuje informaci o spojení. Socket obsadí určitý port (číslo (0 - 65535) představující záhlaví datových bloků, jehož pomocí lze učít pro který program daná data jsou, neboť na jednom počítači – jedné IP adrese, může komunikovat více programů, každý program však bude mít vlastní port) na vybrané adrese nebo se pokusí na vybranou adresu připojit. Po navázání spojení se vytvoří proudy pro odesílání a přijímání dat. Na konci spojení se pak tyto proudy a socket ukončí. K jejich ukončení dojde i při přerušování spojení.

### **3.7 Neveřejná IP adresa**

Neveřejná IP adresa znamená, že na počítači nemůže pracovat aplikace, ke které by se mohli ostatní uživatelé připojovat prostřednictvím Internetu. Jedná se o IP adresu vnitřní

sítě. Ta umožňuje dostupnost daného počítače pouze v síti Intranet nebo v části sítě Internet, která náleží poskytovateli Internetu.

Naopak veřejná IP adresa umožňuje dostupnost počítače z celé sítě Internetu a umožňuje přímé využívání protokolů TCP/UDP.

### **3.8 Možnosti komunikace mezi hráči**

V předchozí kapitole byl vybrán protokol k síťové komunikaci. Dále je třeba upřesnit, jak za jeho použití budou aplikace (hráči) navazovat spojení a komunikovat mezi sebou. Pro tuto potřebu je vhodné vytvořit aplikaci, která z počítače, na němž bude spuštěna, vytvoří server, ke kterému by se hráči připojovali a mohli jeho prostřednictvím zakládat hry. Dále by si připojení hráči mohli nechat vypisovat založené hry, popřípadě by spolu mohli komunikovat prostřednictvím textových zpráv. Za tímto účelem je možné vytvořit servery s různými vlastnostmi.

První možností je aplikace, která vytvoří server pouze pro zprostředkovávání úvodního spojení mezi hráči. Hráč, který by zakládal hru, by odeslal pouze informaci o tom, na jaké adrese je vytvořená hra čekající na připojení protihráčů. Pokud by se do této hry připojil protihráč, navazoval by spojení přímo s aplikací hráče, který hru založil a server by dostal pouze aktualizaci o tom, že se v založené hře změnil počet hráčů. Pokud by pak zakládající hráč zahájil hru, byli by všichni hráči této hry odpojení ze serveru a jeho úlohu by převzala aplikace zakládajícího hráče.

Výhodou takového serveru je, že pouze zprostředkovává hru. Udrží si informace o založených hrách, které filtruje a rozesílá formou výpisů. Server tedy nebude příliš zatížený. Podmínkou pro chod serveru bude jen to, aby byl na veřejně dostupné IP adrese. Tato podmínka bude ale platit i pro zakládající hráče, ke kterým, pokud nebudou mít veřejnou IP adresu, se ostatní nebudou moci připojit. Neveřejnou IP adresu mají často lidé, kteří jsou připojeni přes síť, která pro připojení k dalším sítím sdílí jedno připojení. Vznikl by tak problém, jak řešit připojení těchto uživatelů.

Dalším možným přístupem by mohla být aplikace, která vytvoří server, jenž by zprostředkoval komunikaci mezi hráči. Server by si držel seznam právě připojených uživatelů a obstarával by pak veškerou datovou výměnu mezi hráči. Pokud by hráč založil hru, vznikla by tato hra na straně serveru a protihráč vstupující do hry by tak komunikoval stále pouze se serverem. Server by tedy obstarával datovou výměnu i během hraní hry.

Nevýhodou tohoto serveru by byly vyšší nároky na hardware počítače, na kterém by běžela tato aplikace. Zvýšený objem přenášených dat a intenzita komunikace mezi jednotlivými klienty by na server kladly větší nároky. Požadavek na veřejnou IP adresu by

platil stejně jako u prvního serveru, ale veřejnou IP adresu už by nemuseli mít hráči, kteří zakládají hru. To by bylo užitečné především pokud by server využívali i hráči s alternativními klienty, které by běžely například na PDA nebo na mobilních telefonech využívající JAVU.

### **3.9 Výběr způsobu komunikace mezi hráči**

Na základě úvah v předchozí kapitole byla pro realizaci komunikace mezi hráči zvolena možnost serveru, který by realizoval veškerou komunikaci mezi hráči. Bude si tedy udržovat informaci o připojených hráčích a pokud bude založena hra, server vytvoří pro tuto hru „místnost“, ve které bude mít seznam hráčů, kteří v této místnosti jsou. Pokud dojde k tahu, server tak pošle informaci všem hráčům, které bude mít na seznamu v místnosti. Popřípadě rozešle jinou komunikaci, kterou bude klientský program podporovat

Toto řešení sice zvýší zatížení serveru, nicméně server nebude hráčům vysílat neustále, ale pouze po různě dlouhých časových intervalech, neboť k rozesílání tahů bude docházet až poté, co hráč, který je na tahu, provede svůj tah a odešle ho na server.



## **4 Návrh rozhraní pro zobrazení a ovládání hry**

### **4.1 Požadavky na okno herní aplikace**

Pro aplikaci je nejdůležitější, aby fungovala na co možná nejvíce počítačích s rozumnými hardwarovými požadavky vzhledem k možnostem současné doby.

Předpokládá se, že hra poběží v samostatném oknu, proto je nutné zvolit vhodnou velikost okna. Velikost okna musí být vybrána tak, aby aplikace mohla být provozována na nejrozšířenějších rozlišeních. V dnešní době mezi nejpoužívanější rozlišení patří 1024 x 768, používané hlavně na 17“ monitorech a různá širokoúhlá rozlišení, která používají notebooky.

Z hlediska hardwaru se pak předpokládá počítač s dostatkem operační paměti tj. více než 256MB paměti pod operačním systémem Windows a grafická karta, která umožňuje použití knihovny Microsoft DirectX.

### **4.2 Zpracování okna programu**

Vzhledem k rozšířenosti operačního systému Microsoft Windows, především verze XP, je vhodné zpracovat program tak, aby běžel jako okenní aplikace v tomto prostředí. Pro toto zpracování byly použity formuláře Windows (Windows Forms), které mají vše potřebné pro vytvoření plnohodnotné okenní aplikace. Při teoretických rozvahách bylo uvažováno, že program pro zpracování výstupu pro hráče umožní zobrazení a hraní i jiných deskových her než je „Mlým“, proto je okno opatřeno prvky jako je lišta s menu (umožňuje hráči jednoduše se připojit na server, vybrat hru, založit hru nebo ukončit hru), textové pole, tlačítka (pro zadávání vstupu od uživatele), rozbalovací nabídky (pro výběr prvků, které mají pevně daný rozsah – např. počet hráčů ve hře) a popisky k jednotlivým prvkům. Vzhledem k tomu, že v dnešní době řada uživatelů používá programy měnící výchozí vzhled Windows, použití těchto prvků umožní zachovat celkový vzhled Windows i v této aplikaci. Nezanedbatelnou výhodou je pak také ověřená funkčnost prvků a jejich jednoduché použití.

### **4.3 Zobrazení hry**

Podle úvah v předchozí kapitole bude hra zobrazena ve vlastním okně a hráč uvidí hrací plochu v trojrozměrném zobrazení. Pro trojrozměrné zobrazení je použita perspektivní projekce, kde se vzdálenější objekty jeví jako menší. Další možností by mohla být například

projekce pravoúhlá, kde se objekty jeví jako by byly pozorovány z nekonečně vzdáleného bodu (rozměry bodu nezáleží na jeho umístění). To by mohlo být matoucí, protože lidské oči jsou zvyklé dívat se perspektivně.

Rozvržení mapy při hře „Mlým 3D“ bude vypadat jako mapa na obr. 1 s tím rozdílem, že každý vnitřní čtverec je posunut výše, jako by do pyramidy, viz obr. 4. Při tomto zobrazení je vhodné povolit uživateli naklonění a otočení pohledu, aby v případě potřeby získal lepší vizuální informaci o stavu hry. V okně bude zobrazena informace, jestli je hráč na tahu nebo čeká na tah protihráče. Pokud hráč vyhraje, prohraje nebo odejde ze hry, bude třeba ho opět informovat o této skutečnosti vhodným výpisem. V případě, že hráč nebude mít vybranou žádnou hru, bude zobrazena neutrální obrazovka s prázdnou herní mapou.

Pro trojrozměrné vykreslování hry bude třeba použít programátorskou knihovnu obsahující nástroje pro tvorbu počítačových her nebo jiných multimediálních aplikací. Zde bude probíhat výběr z jedné verze knihoven Microsoft DirectX, které jsou vytvořené firmou Microsoft pro použití pod operačními systémy Windows. Výběr knihovny je ovlivněn její dostupností na současném hardwaru a operačním systémem.

Nejnovější verzí knihovny je DirectX10. Tato verze ale lze použít jen na operačním systému Microsoft Windows Vista, který ale v současnosti ještě není rozšířen. Nevýhodou této verze je také to, že k chodu je vyžadována grafická karta podporující DX10.

Starší verze je DirectX 9.0c, která je dostupná i na operačním systému Microsoft Windows XP a její hardwarová náročnost, vzhledem k dnešnímu běžně používanému hardwaru, není velká. Dostupnost verze DirectX 9.0c se uvádí i u integrovaných grafických karet.

Alternativní zobrazení by mohlo být například pomocí Flash. To by se však hodilo spíše v případě hraní hry ve webovém prohlížeči. Další nevýhodou by bylo obtížně realizovatelné rozšíření aplikace o další deskové hry nebo jiné grafické prvky aplikované pro všechny hry v rámci programu.

Na základě výše uvedených úvah tak byla pro realizaci práce vybrána verze DirectX 9.0c. Tato volba zajistí dostupnost výsledné aplikace na širokém spektru počítačů.

#### **4.4 Ovládání hry**

Ovládání hry by mělo být co nejjednodušší, při ovládání deskových her bude patrně nejvhodnější použít myš, popřípadě doplnit ovládání programu o vhodné klávesové zkratky. Program by také mohl některé činnosti vykonávat automaticky, například pokud hráč dohraje

hru a odejde z místnosti, program může sám vyvolat natažení aktuálního seznamu her, přičemž typ vypsaných her bude určen na základě skončené hry.

#### **4.5 Zobrazení herního serveru**

Jelikož hráč nebude muset přijít do styku s aplikací herního serveru, bude stačit, když herní server bude mít výstup do konzole a jeho ovládání bude prováděno pouze přes klávesnici. Naopak díky tomu bude možné se serverem manipulovat i pokud bude na vzdáleném počítači, ke kterému bude pouze konzolový přístup.

## 5 Další funkce herního klienta a možná rozšíření

### 5.1 Profily a nastavení sítě

Klientská aplikace při prvním spuštění vytvoří složku pro ukládání konfiguračního souboru. V tomto souboru bude uložena IP adresa serveru, port a přezdívka hráče. Aplikace se vždy okamžitě po svém spuštění pokusí najít tento soubor a načíst z něho potřebná data. Pokud soubor nenalezne, pokusí se vytvořit nový soubor s výchozími daty nastavení. Nemůže-li program s tímto souborem pracovat, nebude informace o adrese, portu a profilu uložena a při následujícím spuštění bude hráč muset znovu nastavit adresu serveru a port. Jestliže se aplikaci povede otevřít soubor s nastavením a přečíst z něho údaje, snaží se podle nalezené přezdívky načíst profil hráče ze souboru, který je ve složce ‚Profily‘.

Profil neumožňuje identifikaci konkrétních hráčů, ale jsou zde uloženy dosavadní herní výsledky hráče (skóre) – každá hra podporující profil zde má záznam o počtu uskutečněných her, výher, proher a nedohraných her. Vždy po zahájení hry se zvýší počet odehraných her a her nedohraných. Pokud hráč dohraje standardním způsobem, je zvýšen stav vyhraných nebo prohraných her a zpět snížen počet her nedohraných. V případě, že se hráč během hry odpojí nebo vypne program, zůstane počet nedohraných her zvýšen.

Hráč může v nastavení profilu vytvořit profil s libovolnou přezdívkou nebo si vybrat již existující profil. Profil se po vytvoření nebo výběru nastaví jako výchozí a všechna data pro profil určená jsou do něho ukládána. Hráč si může vytvořit libovolný počet profilů. Ty se nacházejí ve složce ‚Profily‘. Jakýkoli profil může hráč smazat vyjma profilu s přezdívkou ‚noname‘, který je určen jako výchozí a při jeho smazání se profil neodstraní, ale pouze se vynuluje (nastaví se výchozí hodnoty, tj. 0 odehraných her, 0 nedohraných her, apod.).

Program s těmito soubory pracuje i během dalšího běhu aplikace, pokud dojde ke změně nastavení sítě nebo změně profilu. Po dobu, kdy je hráč připojen k serveru, nelze profily mazat, vytvářet ani přepínat, neboť aktuální profil je při připojení načten serverem a protihráčům je dovoleno si tento profil vypsát. Herní server vyžaduje profil pouze při přihlášení hráče, o zasílání aktualizací profilu se během hraní her stará herní klient

V případě, že se nepodaří soubor s profilem otevřít, protože neexistuje nebo je poškozen, pokusí se aplikace tento soubor s profilem znovu vytvořit s vynulovanými hodnotami. Pokud se aplikaci nepodaří soubor vytvořit ani načíst, je na server odeslán profil s vynulovanými hodnotami.

## **5.2 Zvuky**

V programu jsou zvuky, které hráče upozorní na významné události v běhu programu. Konkrétně pokud se úspěšně připojí k serveru, vyhraje hru nebo naopak hru prohraje. Tyto zvuky jsou uloženy ve složce ‚Zvuky‘ (tato složka je součástí složky, ve které je hra) a jsou načteny do programu pokaždé při jeho spuštění. Pro správný chod je třeba, aby měl počítač funkčně nainstalovanou zvukovou kartu. V nastavení je pak možné tyto zvuky vypnout nebo zapnout.

## **6 Možné programovací jazyky pro realizaci programů**

### **6.1 JAVA**

Java je objektově orientovaný programovací jazyk, který vyvinula firma Sun Microsystems. Tento jazyk patří mezi nejpoužívanější jazyky především díky přenositelnosti. Je používán hlavně pro programy, které mají pracovat na různých operačních systémech. Jedná se o jazyk interpretovaný, avšak ztráta výkonu není příliš patrná, neboť překladače zpracovávají jen tu část, která je právě potřeba. Byl navržen pro podporu síťových aplikací a má vlastnosti, které chrání počítač v síťovém prostředí před nebezpečnými operacemi. Také podporuje běh vícevláknových aplikací.

### **6.2 C#**

C Sharp je vysoko úroňový objektově orientovaný programovací jazyk, který vyvinula firma Microsoft. C# vznikl spojením jazyka C a JAVA a lze ho využít jak při tvorbě formulářových aplikací ve Windows, tak pro software pro mobilní zařízení – například PDA a jiné mobilní zařízení pracující s operačním systémem Microsoft Windows Mobile. Jazyk C# je jedním z jazyků, který pracuje s API Manager DirectX

## **7 Realizace síťového rozhraní serveru pro hru „Mlým“**

### **7.1 Herní Server**

Pro realizaci herního serveru byl vybrán programovací jazyk Java, především pro svoji dobrou přenositelnost mezi různými operačními systémy, neboť se dá předpokládat, že serverové aplikace se budou spouštět na různých operačních systémech. Systém komunikace serveru s obsluhujícím uživatelem je realizován přes konzolové výpisy a vstupy. To umožní jednoduchou obsluhu.

### **7.2 Požadavky na herní server**

Jelikož je herní server realizován v programovacím jazyku JAVA, je třeba, aby na serveru byl interpret a překladač JAVA nejlépe ve verzi 1.6.

Pro správný chod serveru je také vyžadováno, aby byl připojený k internetu a měl veřejnou IP adresu. Server je také možno provozovat v počítačové síti bez připojení k internetu, pokud se k serveru budou připojovat hráči v rámci této sítě. Po spuštění server začne poslouchat na vybraném portu. Spojení se realizuje na základě úvah v předchozích kapitolách přes protokol TCP.

### **7.3 Jak se chová server při připojení hráče**

Server okamžitě po spuštění připraví struktury pro uchovávání informací o hráčích a o založených místnostech. Informace jsou uloženy v poli objektů, kde číslo hráče nebo místnosti odpovídá číslu pole.

Server u každého hráče udržuje informace o socketu, což je informace o připojení hráče, o jeho přezdívce, dosaženém skóre, které je předáno z profilu a o tom, jestli je hráč v nějaké místnosti. Tyto informace budou uchovávány v poli objektů. Tuto strukturu pro uložení hráčů program vytvoří hned na začátku po svém spuštění. Velikost tohoto pole také určuje maximální počet hráčů na serveru. Pokud chce jeden hráč hrát ve více místnostech najednou, musí spustit další herní klienty a i s nimi se přihlásit k serveru, přičemž každý klient byt' jednoho hráče je veden jako hráč další.

U místností si server udržuje informace o tom, jestli je v místnosti nějaká hra, o typu hry, která je případně v místnosti, o celkovém a maximálním počtu hráčů (z čehož server

vyhodnocuje, jestli je již v místnosti obsazeno), jestli byla již zahájena hra a informace o hráčích, kteří jsou přítomní v místnosti (ukládá se přezdívka, socket a číslo pole, ve kterém jsou uloženy kompletní informace o hráči). Vzhledem k tomu, že server pracuje s více vlákny a musí tak ke společným datům přistupovat pouze přes synchronizované metody, uložením informací o hráčích v místnosti urychluje proces komunikace s hráči (při zjišťování socketu hráče, kterému se posílá zpráva, se nemusí manipulovat se všemi daty). Maximální počet místností je stejný jako maximální počet hráčů na serveru.

Server byl vytvořen tak, aby umožňoval připojení různých klientských programů. Protože různé klientské programy mohou obsahovat různé typy her a činností, vyzve server každého klienta, který se připojí, aby zaslal svoji verzi. U přijaté verze je ověřeno, jestli se shoduje s verzemi, které server podporuje. Pokud ano, dovolí server klientu dokončit přihlášení.

Během přihlášení je klientský program kromě verze tázán ještě na přezdívku uživatele a případně na skóre uživatele (je načteno z profilu uživatele), které si umí server zapamatovat jako řetězec. Proto pokud klientský program nepodporuje funkci ukládání skóre, může odeslat serveru řetězec se zprávou, že tuto funkci nepodporuje. Pokud pak jiný uživatel požaduje výpis skóre tohoto hráče, server mu tuto zprávu přepošle.

Po úspěšném přihlášení a zapsání hráče do připravené struktury má hráč pomocí příkazů buďto vytvořit vlastní hru nebo realizovat výpis her. Při tvorbě hry odesílá klientský program číselné označení dané hry, její popis a maximální počet hráčů, kteří se hry mohou zúčastnit. Hráč pak čeká ve hře na připojení protihráče. Během čekání může hráč kdykoli z místnosti odejít. Server automaticky zruší místnost, ze které odešel poslední hráč.

Při výpisu hráč posílá serveru číselnou informaci, která určuje, jaká hra bude vypísána. Pokud zašle nulu, je realizován výpis ze všech volných her na serveru.

Pokud klientská aplikace požádá server o výpis dostupných založených her, začne server procházet všechny místnosti a ověřovat, jestli jsou prázdné nebo v nich probíhá hra. Pokud tyto dvě podmínky nejsou splněny a hráč například vypisuje hry ze seznamu všech místností, server před odesláním výpisu spočte kolik existuje místností. V případě, že by mělo být na výpisu více než šedesát místností, server provede náhodný výběr a zmenší počet vypisovaných místností na šedesát.

Jestliže je hráč před místností, má možnost odeslat chatovou zprávu všem ostatním hráčům připojeným k serveru, kteří nejsou v místnosti. Pokud je v místnosti, je realizován chat pouze mezi hráči, kteří jsou v místnosti. V místnosti má také klient možnost odeslat soukromou zprávu libovolnému hráči z této místnosti nebo si nechat vypsat skóre, které hráč poslal na server.



Pokud je hráč v místnosti a do místnosti vstoupí protihráč, může být zahájena hra. O minimálním počtu hráčů pro spuštění hry rozhodují herní pravidla, kterými se pro danou hru řídí herní klient. Server pouze kontroluje počet hráčů místnosti a v případě, že je již v místnosti maximum možných hráčů (je nastavováno při vytváření místnosti) nedovolí dalším hráčům vstup. Po zahájení hry přijde oběma hráčům informace o tom, že hra byla zahájena a vytvářejícímu hráči navíc informace, že je oprávněn provést tah. Pokud hráč provede tah, zasílá se serveru jednak informace o stavu hry, která bude přeposlána ostatním hráčům, ale také informace o tom, jestli bude táhnout ještě jednou nebo má server předat právo tahu dalšímu hráči v místnosti.

Pokud se jeden z hráčů odpojí před koncem hry, zkoumá server počet hráčů, kteří v místnosti zbylí. Pokud v místnosti zůstali alespoň dva hráči, server pokračuje v rozesílání stavu hry s tím, že vynechává odejitého uživatele.

## **7.4 Možnosti serveru**

Při spuštění serveru je vytvořeno jedno vlákno sloužící pro komunikaci formou konzolových vstupů a výstupů. Server umí vypsát: seznam všech hráčů, kteří jsou k němu připojeni (jejich IP adresu, přezdívku a verzi klienta), seznam všech místností (počet hráčů v místnosti a informaci, jestli ve hře probíhá hra), nastavení portu a maximálního počtu hráčů, počet hráčů na serveru a počet místností. Dále umožňuje vypnout / zapnout přijímání nových hráčů na server, zapnout / vypnout vypisování událostí serveru (vypisuje informace o tom, jestli se nějaký uživatel připojil nebo odpojil), rozeslat všem hráčům hromadnou zprávu, která se jim zobrazí formou oznámení nebo vypnout server s tím, že klientským aplikacím rozešle hromadnou zprávu o vypnutí serveru a ukončí spojení.

## 8 Realizace síťového rozhraní pro klientskou aplikaci

### 8.1 Windows klient

Pro realizaci klientského programu byl vybrán programovací jazyk C#. Je předpokládáno, že aplikace bude většinou provozována na počítačích s operačním systémem Windows a je vyžadována podpora Windows Forms a Managed DirectX.

Program obsahuje řešení síťové části pro komunikaci s herním serverem. V programu je vytvořena metoda, která zrealizuje navázaná spojení se serverem a ihned poté vytvoří dvě vlákna. První vlákno bude obstarávat vysílání a druhé přijímání síťové komunikace se serverem. Data pro odesílání budou vkládána do fronty pro odesílání, její stav bude pravidelně kontrolovat vysílací vlákno a příchozí data budou zpracována do fronty tahů nebo do fronty chatu. Těto vlastnosti bude moci být využito při zpracovávání tahu hráče a zpracovávání přijatého řetězce stavu hry. Tím se tak částečně oddělí síťová část aplikace od zbývajících částí zpracovávajících pravidla hry, grafický výstup apod.

V síťové části musí být také obsažen seznam příkazů, kterými server odpovídá - zasílá tahy jiných hráčů nebo chatové zprávy. Jde o zprávy, kterými hráče informuje o založení místnosti, o vstupu nebo odchodu protihráče z místnosti, o spuštění hry. Na vyžádání je možné hráči zaslat informaci o hráčích kteří, kteří jsou připojeni v místnosti a informaci o skóre vybraného hráče.

V této části je třeba také ošetřit chybové stavy, které během komunikace mohou nastat, například ztráta spojení. Dále je třeba řešit chybové stavy serveru nebo provozní zprávy. Chyba serveru může nastat například když hráč odejde z místnost, ale serveru se nepodaří hráče odebrat. Dále je třeba ošetřit správnou reakci na informaci, že je server již zaplněn nebo že již nepřijímá další hráče

## 9 Realizace uživatelského rozhraní a grafického výstupu

### 9.1 Prvky Windows Forms

Pro ovládání programu je využito standardních prvků Windows Forms. Ty jsou použity k výběru hry, k obslužení a nastavení síťového připojení, vypsání seznamu her, popřípadě k odchodu z místnosti a vypsání nápovědy. Tyto prvky jsou přístupné z Menu programu, které je umístěné v klasickém horním pruhu obrazovky. Při připojení k serveru se v dolní části okna objeví prvky umožňující chat mezi uživateli a tlačítka pro zahájení hry nebo odchod z hry.

Pro funkci těchto prvků je potřeba knihoven z Microsoft Framework verze 2.0

### 9.2 Prostorové zobrazení pomocí DirectX

K vykreslení grafického vstupu do trojrozměrného zobrazení je využito knihoven Microsoft DirectX. Jedná se o verzi obsaženou v Microsoft DirectX SDK August 2007 (souprava nástrojů pro vývoj aplikací obsahující dokumentaci a ukázkové příklady). Tato verze je potřebná pro spuštění celé aplikace. Není však třeba stahovat celé SDK, ale jen DirectX End-User Runtime Web Installer. Podrobněji je toto popsáno v příloze B.

Po spuštění programu dochází k inicializaci výše uvedených knihoven a nastavení potřebných parametrů pro DirectX. Dále je v programu vytvořena třída MeshLoad, která se stará o načítání objektů, resp. modelů. Tyto modely lze použít v rámci celého klientského programu všemi implementovanými hrami. Pro práci s objekty je zde i funkce pro jejich vykreslování s nastavením jejich pozice, natočení a případné zmenšení. Všechny objekty jsou v programu uloženy pouze jednou a jsou volány ze struktur pole objektů, tyto struktury jsou inicializovány buďto při spuštění programu nebo krátce před prvním spuštěním vybrané hry a obsahují informace o tom, jaký objekt (mesh) má být vykreslen a na jaké pozici. Tyto struktury slouží také při vybírání prvku z hracího pole po kliknutí hráče na hrací plochu. Toto hledání je realizováno pomocí hledání kolize vektoru myši s vybraným meshem na pozicích, které jsou obsaženy v příslušných strukturách. To umožňuje naklápění hracího pole a výběr prvku bez nutnosti vypočítávání jeho nových souřadnic.

Naklápění hracího pole je dostupné u obou her - Mlýn 2D i Mlýn 3D, které prostorového zobrazení využívají. A to pomocí myši nebo kláves ‚PageUp‘ a ‚PageDown‘. Naklápěcí úhel je 90 stupňů. Hráč si toto naklonění může měnit libovolně během hry,

aby získal lepší představu o stavu hracího pole, pro které je vhodné použít mírně nakloněný pohled. Dále je možné herní mapu otáčet (podle svislé osy procházející středem herní mapy) o libovolný úhel (pomocí pravého tlačítka myši), popřípadě lehce přiblížit nebo oddálit (za stisku prostředního tlačítka myši).

Vykreslovaná scéna se mění podle aktuálního stavu programu. Pokud je vybraná konkrétní hra, tak je také vykreslována, v opačném případě je vykreslována výchozí scéna.

### **9.3 Návody her**

V nabídce menu je možnost vedle síťové hry spustit nápovědu pro konkrétní hru. Zde jsou v několika krocích popsána základní pravidla hry a způsob ovládní doplněn ukázkami na hrací ploše. Mezi jednotlivými kroky je možné se přepínat pomocí mezerníku (o krok dopředu) a klávesy zpět (o krok zpět). Po zobrazení posledního kroku se nápověda ukončí a začne se opět vykreslovat výchozí scéna.

### **9.4 Klient pro PDA**

Použitím programovacího jazyku vznikla možnost zdrojové kódy zcela přenášet nebo s úpravami přepsat pro mobilní zařízení, například PDA. Nevýhodou při přenosu síťové části je absence některých knihovních funkcí v síťové komunikaci. Ty musí být nahrazeny jinými funkcemi nebo být zčásti zcela napsány. I přes tuto komplikaci je však přenos síťové části relativně bezproblémový.

Dále nebylo obtížné přenést funkce starající se o kontrolu stavu hry a o dodržování pravidel. Zbývalo jen doplnit uživatelské rozhraní pro hraní hry. Pro řešení hry 2D Mlým byl na formulář aplikace natažen obrázek s příslušnou mapou a rozmístěna tlačítka tak, aby nahradila plošky pole, na kterých je možné manipulovat s kameny.

## 10 Sumarizace požadavků na HW a SW počítače

### 10.1 Server

Pro plnohodnotné využití klientských aplikací je třeba spustit server na počítači s veřejně dostupnou IP adresou. Alternativou může být spuštění serveru v rámci počítačové sítě, ve které bude dostupný ostatním uživatelům sítě. Pro spuštění aplikace na serveru je potřeba překladač a interpret jazyka JAVA ve verzi alespoň 1.6\_0\_05<sup>2</sup>.

### 10.2 Windows klient

Při použití Windows klienta je třeba mít nainstalovaný Framework verze 3.5<sup>3</sup>, ze kterého byly použity některé knihovny. Také je třeba mít knihovny Managed DirectX<sup>4</sup>, které jsou použity pro trojrozměrný výstup.

Rovněž je potřeba vhodný hardware, konkrétně grafická karta podporující Microsoft DirectX 9.0c a zvuková karta pro přehrávání zvuků.

### 10.3 PDA klient

Zařízení mobilního klienta musí mít operační systém Microsoft Windows Mobile 5 a vyšší. Dále je třeba funkční spojení se sítí internet nebo jinou sítí, prostřednictvím které by se herní klient mohl připojit k serveru.

---

<sup>2</sup> Je podrobně popsáno v příloze B.

<sup>3</sup> Je podrobně popsáno v příloze B.

<sup>4</sup> Je podrobně popsáno v příloze B.

## **11 Získané poznatky a ověření funkčnosti**

### **11.1 Získané poznatky**

Při řešení jsem nabyl cenných informací o rodině protokolu TCP/IP. Tyto informace jsem poté také prakticky využil při tvorbě serverové aplikace a klientských programů. Především při tvorbě serveru jsem pak měl možnost prozkoumat práci vláken, která přistupují ke stejným datům. Další zajímavou zkušeností bylo vytváření trojrozměrného zobrazení hrací plochy mlýn, které jsem následně doplnil o nové tahy ve vzniklém pyramidovém mlýnu.

### **11.2 Ověření funkčnosti**

Funkčnost programů byla vyzkoušena během hraní hry Mlýn2D a Mlýn3D, přičemž jsem se snažil odstranit všechny chyby programu, které by se mohli objevit a zavést opatření, která by těmto chybám zabránila.

Jednotlivé metody programu byly zkoušeny připojením klienta s konzolovým rozhraní, aby mohlo být ověřeno zda-li se chovají očekávaným způsobem.

Testována byla zejména práce vláken na herním serveru, který by měl být schopný obsloužit připojení i většího množství hráčů. Pro prověřování výkonu serveru je na serveru funkce, která zobrazí za jaký čas server vyřídí došlý příkaz od klienta včetně odeslání případných odpovědí. Tyto časy byli při připojení zhruba 20 klientských aplikací dostatečně malé (desetiny milisekund) a dá se proto předpokládat, že server by měl bez problémů zvládnout připojení přednastavených 200 hráčů.

U klientských programů pak bylo prováděno testování hraním hry mezi jednotlivými klienty přes server, který byl umístěn na localhostu, lokální síti i internetu.

## 12 Závěr

V rámci bakalářské práce jsem se seznámil s pravidly deskové hry mlýn. Analyzoval různé modifikace, rozdíly v pravidlech, které využívají jiné národy a navrhl pravidla pro deskovou hru Mlýn 2D a Mlýn 3D

Byl vytvořen herní server umožňující síťovou hru v podstatě libovolné deskové hry pro dva a více hráčů. Tento server realizuje veškerou komunikaci mezi hráči připojenými na server, proto zde může bez problémů založit hru jak hráč, který se se svým klientem připojuje s neveřejnou IP adresou, aniž by se musel snažit skutečnost neveřejné adresy nějak ošetřovat.

Dále byl vytvořen Windows klient, který tento server využívá pro hraní hry Mlýn a její modifikace. V klientském programu byla vytvořena síťová část tak, aby ji bylo možné využívat i dalšími hrami, které by se mohly dodatečně do programu naimplementovat.

Klientský program by realizován pomocí jazyku C# a mohl tak být částečně přenesen do klientské aplikace fungující na mobilních komunikačních zařízeních.

Rozhraní mezi hráčem a klientským programem bylo realizováno přes grafické prostředí umožňující trojrozměrné zobrazení hrací plochy.

## Literatura

- [1] DOSTÁLEK, Libor – KABELOVÁ, Alena . *Velký průvodce protokoly TCP/IP a systémy DNS*. 3. aktualizované a rozšířené vydání. Praha: Computer Press, 2002, ISBN 80-7226-675-6.
- [2] Mlýn (hra) [online]. c1999 Poslední revize 2. 2. 2008 [cit.2008-3-4] Dostupné z URL: <[http://cs.wikipedia.org/wiki/Ml%C3%BDn\\_%28hra%29](http://cs.wikipedia.org/wiki/Ml%C3%BDn_%28hra%29)>
- [3] HEROUT, Pavel. *Učebnice jazyka JAVA*. 2. vydání. České Budějovice: KOPP, 2006, ISBN 80-7232-115-3.
- [4] SHARP, John – JAGGER, Jon. *Microsoft Visual C# .NET krok za krokem*. Přeložil Jan Gregor. 1. vydání. Praha: Knihy Idnes, 2002, ISBN 80-86593-27-4.
- [5] PETZOLD, Charles. *Programování Microsoft Windows v jazyce C#*. Přeložil Jan Pokorný, Pavel Vaida. Praha: SoftPress, 2003, ISBN 80-86497-54-2.
- [6] MILLER, Tom. *Programujeme 3D hry v jazyce C#*. Přeložil Ondřej Baše. Brno: Computer Press, 2006, ISBN 80-251-1126-1.



## Příloha A – Uživatelská dokumentace ovládání

### Ovládání serveru

Zkompilovaný server spustíme z příkazové řádky příkazem *java StartGameServer*<sup>5</sup>.

Po spuštění serveru bude uživatel vyzván k vyplnění portu, na kterém bude server a klientské aplikace komunikovat. Jako číslo portu se doporučuje volit dostatečně vysoké číslo. Pro spuštění serveru na desktopovém počítači s operačním systémem Windows by nemělo být vyžadováno administrátorské přihlášení. Pokud by ale aplikace nešla spustit, je třeba v administrátorském přihlášení povolit aplikaci komunikaci na vybraném portu. Nižší porty jsou vyhrazeny aplikacím, například port 80 pro webové stránky.

Po úspěšném spuštění serveru může uživatel vypsat nápovědu stisknutím tlačítka ‚h‘. Poté se mu zobrazí veškeré možnosti herního serveru, viz obr. 5. Pro volbu:

‚1‘ - vypnutí / zapnutí vypisování herních událostí.

‚2‘ – vypnutí/zapnutí přijímání nových hráčů na server. Nově přistupujícím hráčům bude vypsána

hláška, že server v tuto chvíli nepřijímá další hráče.

‚p‘ – vypíše počet hráčů připojených na server a počet hráčů, kteří jsou v nějaké místnosti

‚l‘ – vypíše číslo hráče, jeho IP adresu, přezdívku a verzi klienta

‚n‘ – vypíše jaký je na serveru nastaven port a kolik hráčů je připojeno z celkového počtu možných

‚m‘ – vypíše všechny místnosti, ve kterých je nějaký hráč, s informací jestli je spuštěná hra

‚k‘ – po této volbě bude uživatel vyzván, aby vložil řetězec, který bude přeposlán všem hráčům na serveru. Tuto funkci je vhodné použít pro informování hráčů, že server bude odstaven

‚o‘ – zapnutí/vypnutí vypisování doby, po kterou hernímu serveru trvalo, než vyřídil došlý příkaz od klientské aplikace

‚E‘ – (velké E) rozešle všem hráčům zprávu, že se server vypíná, ukončí s nimi spojení a serverová aplikace se vypne.

---

<sup>5</sup> Jak zkompilovat server je podrobně popsáno v příloze B.

```

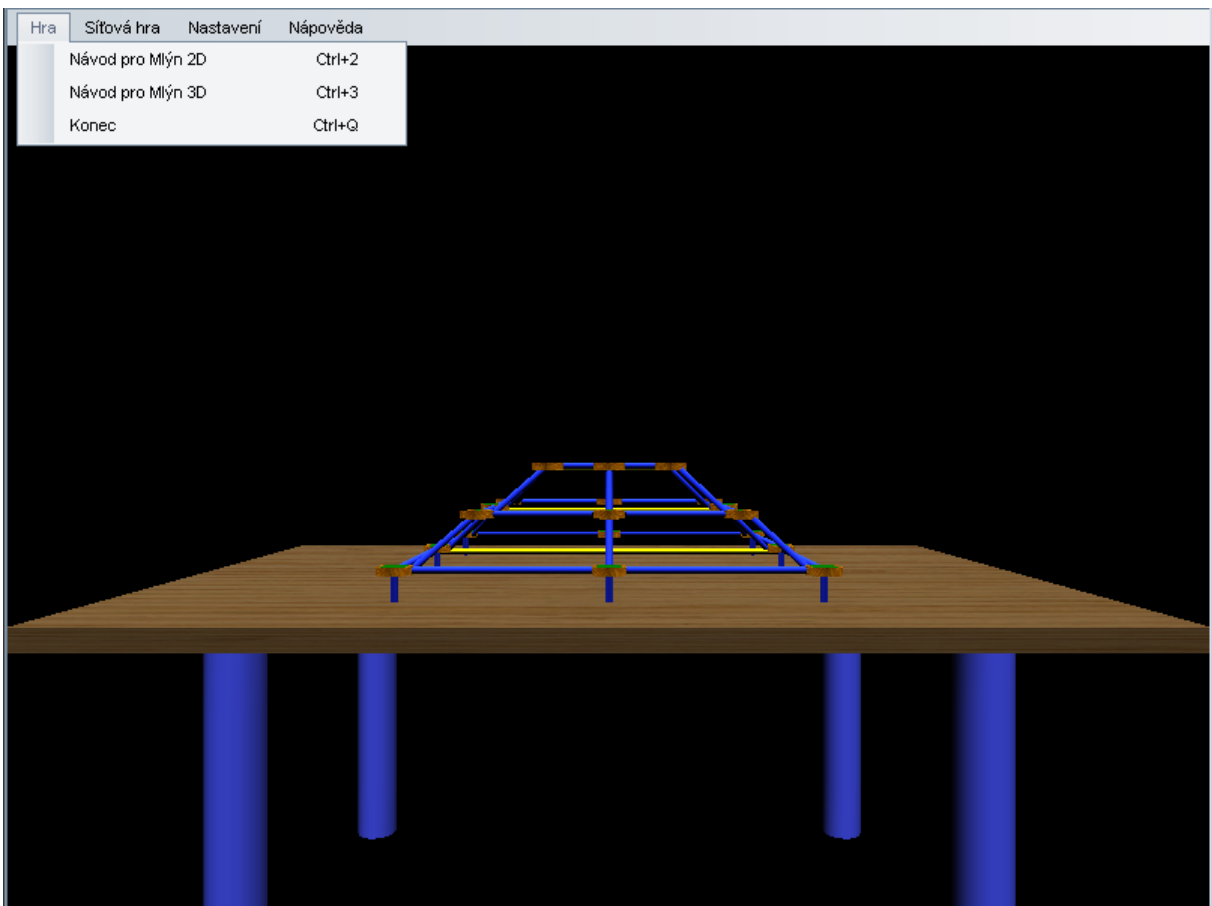
Zadejte port:
8111
Utvoreni socketu...
Server bude komunikovat na portu 8111
Pro napovedu stisknete 'h'
h
+
:
:           Napoveda
:
: 1 - zapnout/vypnout vypisovani udalosti serveru
: 2 - zapnout/vypnout prijimani novych hracu na server
: o - zapnout/vypnout vypisovani casu pro zpracovani prikazu
: p - vypsat pocet hracu na serveru a pocet hracu v mistnosti
: l - vypise cislo hrace, jeho ip a prezdivku
: n - vypise nastaveni serveru (port, maximalni pocet hracu)
: m - vypise stav mistnosti
: k - rozesle zpravu formou messageboxu, dle zadaneho textu vsem hracum
: E - vypne server

```

Obrázek 5: Zobrazení serveru v příkazové řádce s vypsanou nápovědou

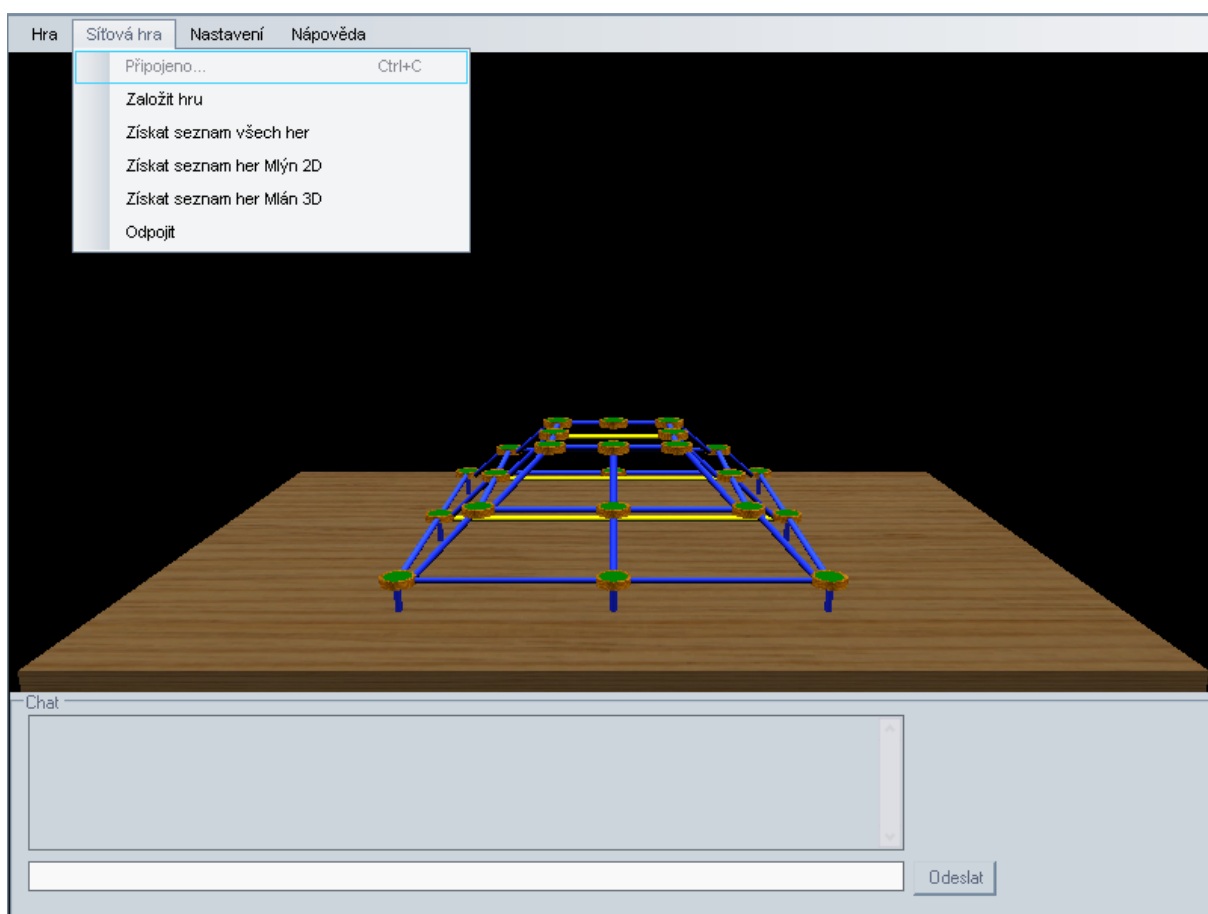
## Ovládání Windows klienta

Po spuštění okna klienta se hráči zobrazí základní nabídka dostupná v offline stavu programu viz. obr. 6. Hráč má možnost přes položku v menu ‚Hra‘ vybrat návod k dostupné hře nebo ‚Konec‘ pro ukončení aplikace. Pokud vybere položku ‚Návod pro...‘ budou mu v několika krocích vysvětlena základní pravidla hra a způsob ovládání hry. V další položce menu ‚Síťová hra‘ (obr. 7) – se hráč připojí k serveru. Při úspěšném připojení k serveru se hráči zobrazí chatový panel a v menu v položce ‚Síťová hra‘ se odemknou všechny možnosti.

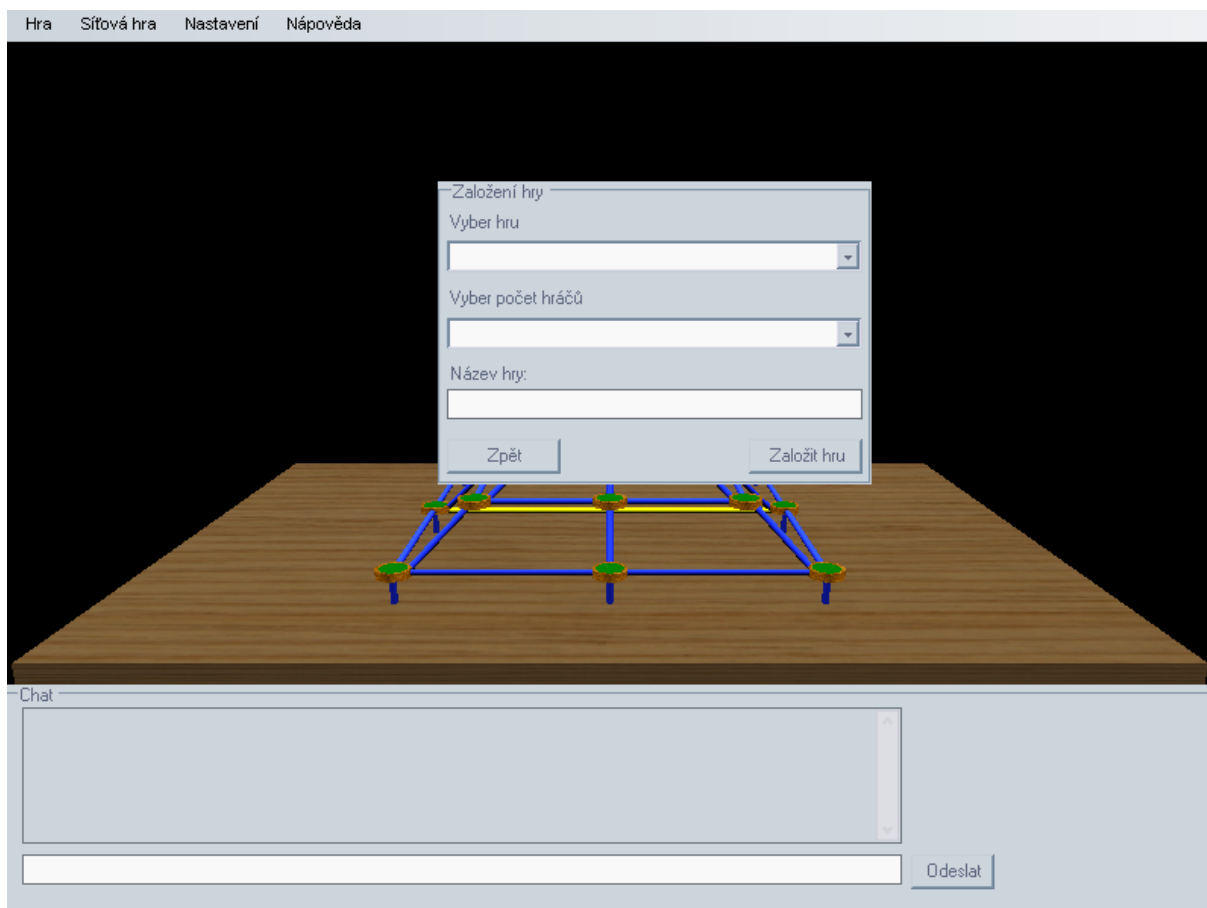


Obrázek 6: Okno clientského programu po spuštění

Po připojení se k serveru (viz obr. 7) může hráč založit hru, což provede vyplněním příslušné tabulky (viz obr. 8), kde vybere hru, zvolí počet hráčů a vyplní název hry. Lze také provést výpis her – a to buď všech her, které jsou na serveru nebo vybrat výpis pouze konkrétní hry. Například stiskem tlačítka ‚Získat seznam her Mlýn 3D‘ obdrží hráč pouze seznam her Mlýn 3D, ve kterých čeká zakládající hráč na protihráče. Výpis se skládá ze dvou oken zobrazených v hlavním formuláři. Levé okno obsahuje seznam her a napravo je informace o tom, jaká se v místnosti nachází hra, kdo ji založil a kolik je tam právě připojených hráčů. Poslední položkou v nabídce ‚Síťová hra‘ je položka ‚Odpojit‘.



Obrázek 7: Stav programu po připojení k serveru

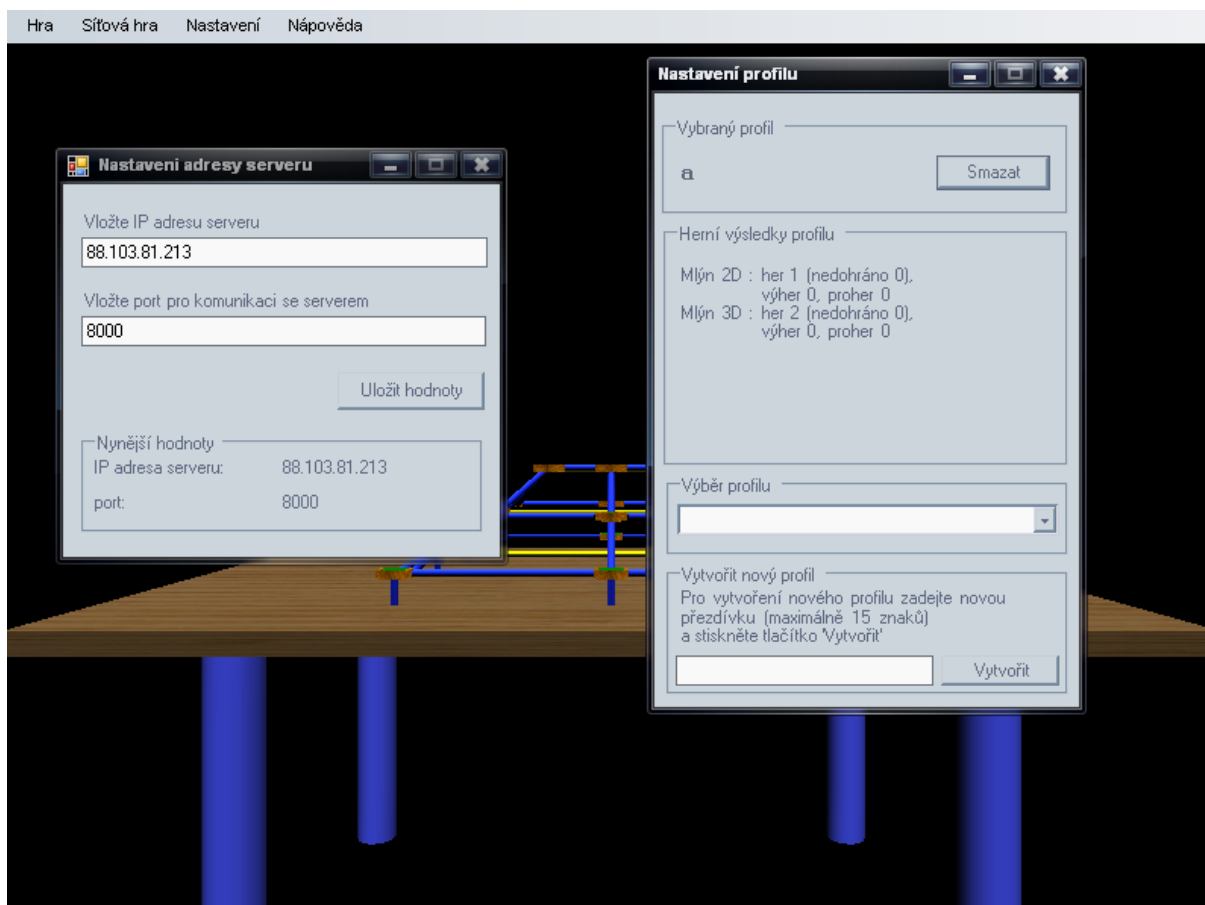


Obrázek 8: Tabulka pro vytvoření místnosti.

Pokud hráč vstoupí do místnosti jako zakládající, čeká na připojení protihráče. Když se protihráč připojí, objeví se zakládajícímu hráči v dolním chatovém panelu tlačítko ‚Start hry‘, které spustí danou hru. Poté probíhá hra na základě pravidel určených v návodu pro konkrétní hru. Časový limit pro provedení tahu není nastaven.

Hráč, který je v místnosti, má na chatovém panelu následující možnosti a výpisy. Vidí tlačítko odeslat, kterým může poslat zprávu protihráčům a seznam uživatelů v místnosti. Každý uživatel má za svojí přezdívkou na tomto seznamu číslo. Díky tomuto číslu můžete hráči v této místnosti poslat skrytě zprávu nebo si nechat vypsát jeho skóre. Po skončení hry server čeká až z místnosti odejde poslední hráč. Poté místnost zruší.

Další položkou v horním menu hry je ‚Nastavení‘ - obsahuje možnost povolit nebo vypnout zvuky, ‚Nastavení sítě‘ (umožňující měnit port a IP adresu herního serveru) a ‚Profily‘. V položce profily si může hráč vytvořit nový profil, popřípadě prohlížet skóre některého ze současných profilů, viz obr. 9.



Obrázek 9: Zobrazení oken pro nastavení sítě a profilu.

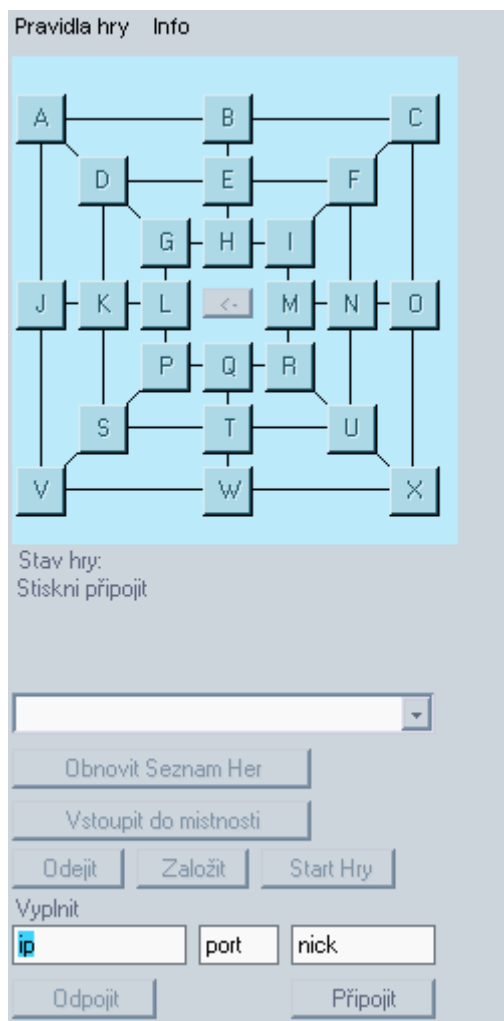
Poslední položkou menu je nabídka ‚Help‘ obsahující nabídku ‚Nápověda‘, jak ovládat program klienta a položku ‚O Programu‘. Volba položky ‚O Programu‘ vytvoří okno s textem, kdy byl program vytvořen a v jakém jazyce byl napsán.

Celý program se ovládá myší, popřípadě částečně klávesnicí. Pomocí levého tlačítka jsou vybírány herní kameny a je určováno, na jakou pozici se mají přesunout. Při stisknutí prostředního tlačítka myši je možné mírně oddálit nebo přiblížit herní pole (stejně tak stisknutím kláves ‚Home‘ a ‚End‘). Stisknutím pravého tlačítka myši a pohybu směrem nahoru a dolů je možné naklápět herní mapu o devadesát stupňů (nebo tlačítka ‚Page Up‘ a ‚Page Down‘), při pohybu do stran se herní mapa otáčí o libovolný úhel.

## **PDA klient**

PDA klient je zjednodušenou verzí Windows klienta (obr. 10). Umožňuje hrát pouze hru Mlýn 2D.

Menu obsahuje položky ‚Pravidla hry‘ – vybráním se vytvoří okno, ve kterém budou vypsána pravidla hry. Dále položku ‚Info‘, pod kterou se skrývají nabídky ‚Ovládání‘ (popisuje obsluhu programu) a ‚O Programu‘ obsahuje informace o programu – na jaké verzi klienta funguje, v jakém jazyce byl vytvořen apod.



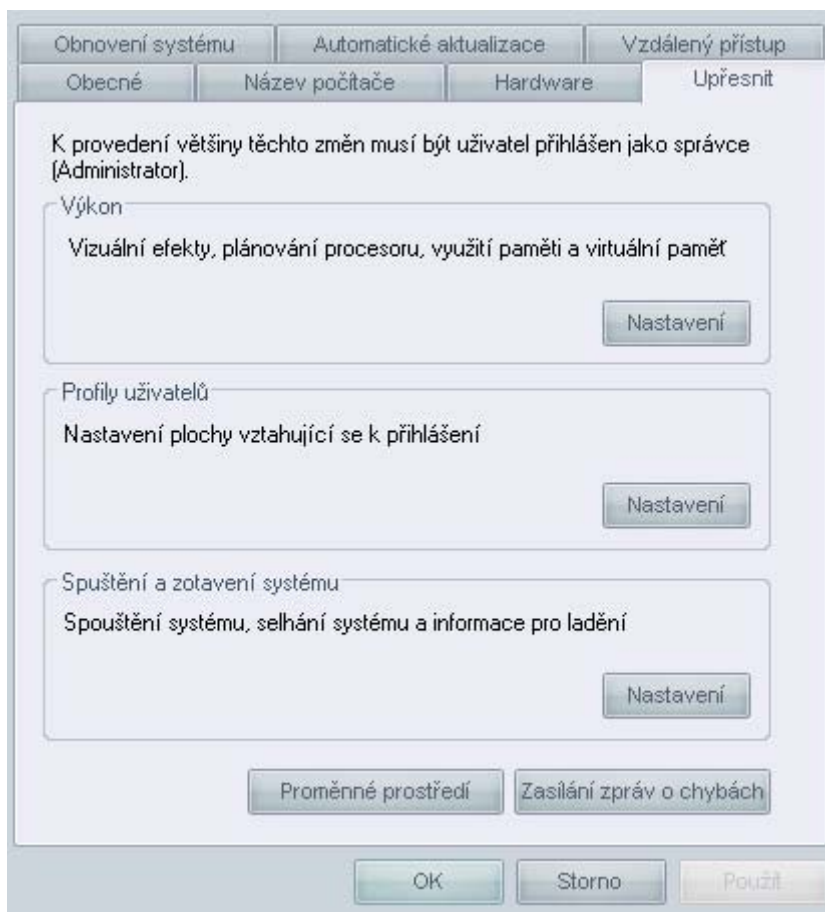
Obrázek 10: Vzhled aplikace pro hru Mlýn 2D na PDA

## Příloha B – popis instalace

### Server

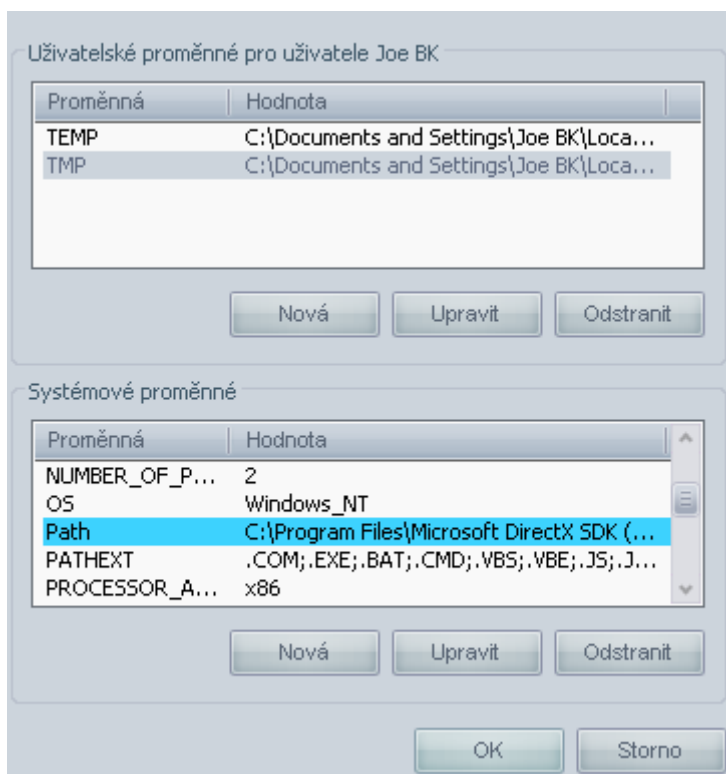
Aby bylo možné rozběhnout server, je třeba mít na počítači nainstalovaný překladač a interpret Javy (tyto komponenty jsou souhrnně označovány jako JDK) . Ověření dostupnosti těchto komponent provedeme napsáním příkazu „javac“ do příkazové řádky. Pokud je interpret a překladač Javy nainstalovaný, objeví se výpis možných parametrů pro tento příkaz a program pro herní server je možno spustit.

V případě, že Java na počítači nainstalována není, je třeba přejít na adresu oficiálních stránek Sun Microsystems <http://java.sun.com/javase/downloads/index.jsp> a stáhnout JDK, například verzi „JDK 6 Update 6“. Před samotným stažením je uživatel vyzván, aby vybral operační systém, na kterém bude JDK provozováno a jazyk. Po stažení se provede standardní instalace, ve které bude uživatel dotázán, kam bude Java nainstalována (doporučené je nechat vše tak, jak je implicitně nastaveno). Pro instalaci Javy je třeba použít administrátorský účet neboť účet s omezeným oprávněním instalaci nedovoluje. Po nainstalování Javy je ještě třeba nastavit cesty k souborům překladače a interpreta. To se provede kliknutím pravého tlačítka na položku „Tento počítač“, která je umístěná na pracovní ploše, popřípadě v nabídce „Start“. V zobrazené nabídce vybereme položku „Vlastnosti“. (Jiným způsobem pro otevření následujícího dialogového okna je přes „Ovládací panely“ vybráním položky „Systém“.) Volbou položky „Vlastnosti“ se zobrazí dialogové okno s několika záložkami. Zde je třeba zvolit záložku „Upřesnit“ a na ní tlačítko „Proměnné prostředí“ viz obr. 11



Obrázek 11: Dialogové okno – Vlastnosti systému

Poté se zobrazí další dialogové okno, ve kterém se nastavují uživatelské a systémové proměnné viz obr. 12.



Obrázek 12: Dialogové okno – Proměnné prostředí



Je potřeba provést úpravu v systémových proměnných. Vybere se proto proměnná „Path“, její hodnoty lze upravovat stisknutím tlačítka „Upravit“. Tím je vyvoláno další dialogové okno, které obsahuje název a hodnotu proměnné. V kolonce pro hodnotu proměnné je již pravděpodobně uvedeno několik cest pro jiné aplikace. Tyto cesty je třeba ponechat nezměněné. Pouze na konce je nutné napsat středník, pokud tam již není a doplnit cestu k adresářové struktuře, ve které je umístěn soubor java.exe. (Pokud během instalace bylo ponecháno implicitní nastavení, bude hodnota vypadat následovně.)

```
C:\Program Files\Java\jdk1.6.0_06\bin;
```

Po stisknutí tlačítka „OK“ se dialogové okno zavře a provedená změna se uloží do systému. Tlačítkem „OK“ se pak potvrdí tato změna ve všech otevřených dialogových oknech. Pokud vše funguje, vypíše systém po napsání příkazu „javac“ nápovědu tohoto příkazu.

Pokud by se stalo, že uživatel nemá přístupová práva pro měnění systémových proměnných, musí být upravena proměnná „Path“ (v dialogovém okně „Proměnné prostředí“ v horní části okna, tj. v „Uživatelské proměnné“ viz. obr. 12. Pokud by tam tato proměnná nebyla, vytvořila by se nová stisknutím tlačítka „Nová“.

Pokud je na počítači nainstalována Java, je pro správný běh aplikace potřeba zkopírovat všechny soubory do jedné složky na disk počítače, kde bude server spuštěn. Příklad souborů \*.java se provede v příkazovém řádku příkazem „javac StartGameServer.java“ a po přeložení se spustí příkazem „java StartGameServer“. Při zadávání obou příkazů je třeba, aby aktuální cesta příkazového řádku byla ve složce, ve které se soubory serveru nachází. Příklad a spuštění aplikace po nainstalování Javy lze také provést pomocí souboru „startWIN.bat“, který je umístěn ve složce se soubory serveru. Tím se spustí příkazový řádek s příkazy pro překlad a spuštění serveru.

V případě, že se překlad a spuštění aplikace nezdaří, je možné, že je na počítači nainstalovaná zastaralá verze Javy nebo byla špatně vložena proměnná „Path“. V tomto případě je třeba odinstalovat starou verzi Javy a nainstalovat novou nebo zkontrolovat správnost nastavení proměnné „Path“.

## ***Klient WIN***

Pro bezproblémové spuštění a chod klientského programu zkopírujte celou složku s programem na disk a spusťte souborem BPmain.exe. Pokud aplikace nelze spustit nebo po startu dojde k jejímu pádu, ověřte, zda máte nejnovější verzi Microsoft .NET Framework 3.5 a knihovny Managed DirectX verze 1.0.2902.0.

Pokud chybí .NET Framework 3.5, lze jej stáhnout z adresy [www.microsoft.com/](http://www.microsoft.com/). Pro jeho instalaci pro Windows XP je vyžadováno, aby byl v systému nainstalován Service Pack 2 a Microsoft Windows Installer 3.1. Vše lze nalézt na adrese [www.microsoft.com/](http://www.microsoft.com/).

Pro získání knihoven Managed DirectX požadované verze je třeba z adresy [www.microsoft.com/](http://www.microsoft.com/) stáhnout „DirectX End-User Runtime Web Installer“, který zajistí stažení a nainstalování potřebných knihoven.

Pro nainstalování .NET Framework 3.5, Service Pack 2, Microsoft Windows Installer 3.1 a DirectX End-User Runtime Web Installer je třeba, aby byl uživatel přihlášen pod administrátorským účtem.

Dále je potřeba mít nainstalované příslušné ovladače grafické karty. (Pro nejrozšířenější typy grafických karet jsou ovladače ke stažení na [www.nvidia.com/](http://www.nvidia.com/) (pro karty NVIDIA) a [www.amd.com/](http://www.amd.com/) (pro karty ATi).)

## ***Klient PDA***

Pro chod mobilního klienta je zapotřebí mobilní komunikační zařízení s operačním systémem Microsoft Windows Mobile 5.0.

## Příloha C – Programová dokumentace

### **Herní server**

Jako první se na serveru spustí ve třídě ‚StartGameServer‘ metoda main, v ní se připraví datová struktura pro ukládání hráčů a pro ukládání založených místností. Pro hráče i místnosti je to pole objektů.

Objekt (Hrac) v poli hráčů pracuje s následujícími proměnnými: Přezdívka hráče, číslo hráče, verze klientského programu, socket a skóre. Tento objekt dále obsahuje statické synchronní metody pro vložení hráče do seznamu, odebrání hráče ze seznamu, vypsání počtu hráčů a metody pro odpojení hráče serverem.

Objekt z pole místnosti (Room) uchovává jméno místnosti, číslo místnosti, maximální počet hráčů v místnosti, typ hry, počet připojených hráčů, informaci jestli v místnosti probíhá hra, číslo hráče, který je na tahu a seznam objektů („hraciVMistnosti“), kde je uchován seznam aktuálních hráčů v místnosti (z tohoto seznamu je umožněno rychle vybírat socket při odesílání řetězce hráči).

Po inicializaci datových struktur uvedených v předchozích odstavcích dojde ke spuštění vlákna z třídy ‚OvladaniServeru‘, kde je možné zapínat různé výpisy a měnit některá nastavení serveru. Podrobný seznam je uveden v příloze A.

Dále je v metodě main spuštěna smyčka, kde se vytvoří spojení s přístupujícími hráči – server začne očekávat klienty na zadaném portu. Pokud se nějaký klient připojí, server pro něj vytvoří vlákno, kde bude klient vyzván, aby vložil svoji verzi programu, přezdívku a skóre. Pokud tak neučiní, nebude vpuštěn do herní smyčky. V opačném případě vstoupí do herní smyčky, kde bude server čekat na jím odeslané zprávy a ty bude v rámci jeho vlákna zpracovávat.

Poslední důležitou třídou herního serveru je třída ‚RoomManipulace‘. Zde jsou prováděny manipulace se seznamem místností. Tento seznam stejně jako seznam hráčů sdílí všechna vlákna. Je tedy třeba k němu opět přistupovat synchronizovaně, aby nedošlo ke ztrátě dat.

Jednotlivé funkce pro práci uvnitř místnosti jsou ve třídě ‚Room‘.

Většina komunikace se serverem je prováděna přes řetězce, které začínají kombinací sedmi domluvených znaků.

Server tedy při komunikaci s herním klientem vysílá tyto příkazy:

GIVEVER – žádost o poslání verze klienta, odpovídá VERISOK nebo VERISBD

GIVENAM – očekává přezdívku hráče, pokud je v pořádku, odešle hráči přidělené číslo,

jinak odešle ERR0001

GIVESCRCR – žádost o výpis hráčova skóre

WELLCOM – pokud se povede vložit hráče do připravené struktury a získá pro něj vlákno

ROOMNWS – informuje klientský program o vstupu protihráče

LEAVE – z místnosti odešel protihráč

XLEAVE – z místnosti odešel zakládající hráč a místnost tak bude zrušena

SERVER\_CLOSE – odesílá, připojujícím se hráčům, pokud má zastavené přijímání dalších hráčů

SERVER\_FULL – odesílá, pokud již není možné z důvodu kapacity serveru přijímat další hráče

Dále server umí odpovídat na tyto příkazy od klienta:

ROOMCRT:maximum hráčů v místnosti:typ hry:jméno hry

-příkaz na založení hry, že zakládající hráč je do ní automaticky vložen a je mu odesláno potvrzení RMCRTOK, jestli byla místnost založena nebo RMCRTNO pokud nebyla  
ROOMVYP:typ hry – vypíše seznam her na serveru, vrací řetězec, ve kterém je číslo místnosti, jméno zakládajícího, jméno hry, typ hry, počet přihlášených hráčů a maximum možných hráčů

ROOMVST:číslo místnosti – klient, který tuto zprávu odeslal, se pokouší vstoupit do místnosti

ROOMIFP – po přijetí server odešle klientovi seznam hráčů v místnosti

ROOMNFO číslo hráče – server odešle skóre vybraného hráče klientovi

ROOMPMS:číslo hráče: zpráva – odešle zprávu vybranému uživateli v místnosti

ROOMLVE – odejde z aktuální místnosti (pokud v nějaké je)

ROOMGPL – tento řetězec může akceptovatelně odeslat pouze hráč, který založil hru – znamená to zahájení hry v místnosti (server tuto místnost nebude vypisovat a nebude do ní vpouštět nové hráče), na tento řetězec reaguje server rozesláním řetězce RMGSTRT (hra byla zahájena) a následuje řetězec RMGTAH1 – tímto řetězcem se předává zakládajícímu hráči právo tahu. Na to hráč odpovídá buďto řetězcem RMNTAH2:0:stav hry – pokud bude ještě táhnout nebo RMNTAH2:1:stav hry – pokud chce povolit serveru předání tahu dalšímu hráči.

ROOMMSG:text zprávy – rozešle přijatou zprávu všem hráčům v místnosti kde je i hráč, který tuto zprávu poslal

MSGBFRM:text zprávy – rozešle zprávu všem hráčům, kteří nejsou v místnosti

Pokud server rozesílá hromadnou zprávu všem hráčům, použije znaky SRVOVER a následuje zpráva.

## ***Klient Windows***

Hlavní třídou pro Windows klienta je třída Program.cs – zde je vytvářeno hlavní okno herní aplikace, probíhá zde inicializace knihoven Managed DirectX a jsou zde definované všechny prvky menu a jejich funkce včetně funkce připojit. V této funkci se vytváří síťové připojení. Pokud se úspěšně naváže, přejde síťové připojení do dvou vláken. Vlákno ListenThread() se ve své smyčce stará o všechny příchozí řetězce a zpracovává je podle typu prvních 7 znaků. Když se jedná o zprávy zasílající tah, je tato zpráva uložena do fronty pro seznam tahů, jinak je uložena do fronty pro zobrazení chatu. Pro odesílání hry funguje vlákno VysilaciVlakno(), které neustále kontroluje jestli není ve frontě na odeslání nějaký řetězec. Všechny použité fronty jsou synchronizované, nemůže tak dojít k narušení dat.

Třída Program.cs dále obsahuje metodu Render(), která se stará o vykreslování herních dat. V této metodě však není přímo renderování objektů, ale volají se pouze metody tender ze tříd MeshLoad nebo SpriteLoad. Metody MeshLoad slouží k načítání objektů a jejich vykreslování na zadaných pozicích, SpriteLoad je podobný metodě MeshLoad, ale místo objektů umí načítat obrázky jako Sprite. Jednoduchou modifikací této třídy pak lze udělat i načítání a vykreslování animovaných Spritů.

Herní data jsou pak uchováována ve třídách Mlyn2D.cs Mlyn3D.cs, kde jsou definována potřebná pravidla pro tyto hry a seznam pozic, na které se mají vykreslovat jednotlivé grafické prvky. Herní data jsou opět připravená jako pole objektů, protože prvků, které je třeba vykreslit je konečný počet. Vlastnosti objektů jsou definovány ve třídě Objekty.cs.

Projekt také obsahuje „Resources File“ – lang.resx, ve kterém jsou uloženy všechny textové výstupy herního klienta. Jednoduchou modifikací tohoto souboru je tak možné program přeložit např. do anglického jazyka.

Pro uchovávání stavu hry, o tom jestli je hráč ve hře nebo ne apod. je použita třída Hrac.cs. Dále jsou použity knihovny DLLmain (obsahuje třídu pro debugovací výpisy), DLLnastavení (obsahuje třídu pro práci se souborem nastavení) a DLLprofil (zde jsou zpracovány metody pro práci s profily).

## ***PDA klient***

PDA klient funguje po síťové stránce obdobně jako klient Windows. Liší se v práci s herními daty. Ta se zpracovávají ve třídě Hrac.cs, kde je také uložena informace o stavu hry. Pro aktualizaci prvků zobrazujících hrací pole je přítomna metoda Prekresli(), která je volána přijímacím vláknem nebo po stisknutí tlačítka na hrací ploše.