

# Posudek bakalářské práce

Matematicko-fyzikální fakulta Univerzity Karlovy

**Autor práce** Klára Cihlářová  
**Název práce** Implementace hry Abaku  
**Rok odevzdání** 2022  
**Studijní program** Informatika  
**Studijní obor** IPSS

**Autor posudku** Jan Kofroň Oponent  
**Pracoviště** Katedra distribuovaných a spolehlivých systémů

## K celé práci

lepší OK horší nevyhovuje

	lepší	OK	horší	nevyhovuje
Obtížnost zadání		X	X	
Splnění zadání			X	
Rozsah práce <i>... textová i implementační část, zohlednění náročnosti</i>			X	X
<p>Zadání práce je celkově poměrně jednoduché, její provedení, a to jak v kvalitě a rozsahu textové části, tak implementace, minimalistické. Celkově působí spíše dojmem jednoduššího zápočtového programu, implementaci se mi nepodařilo plně otestovat (více dále), kvalita textové části nedosahuje očekávané úrovně na MFF UK (více opět dále).</p>				

## Textová část práce

lepší OK horší nevyhovuje

	lepší	OK	horší	nevyhovuje
Formální úprava <i>... jazyková úroveň, typografická úroveň, citace</i>			X	
Struktura textu <i>... kontext, cíle, analýza, návrh, vyhodnocení, úroveň detailu</i>		X		
Analýza		X	X	
Vývojová dokumentace		X	X	
Uživatelská dokumentace		X		

Text je psán česky, obsahuje malé množství gramatických chyb, typografie a struktura textu jsou v pořádku. Jazyk práce je poněkud neformální, autorka přechází od první osoby množného čísla do jednotného, na řadě míst chybí interpunkce. Problematický je nicméně zejména obsah. Po krátkém úvodu a popisu pravidel hry následuje část Analýza, která sice místy diskutuje alternativy řešení různých aspektů programu, nicméně zvolené alternativy se nejeví jako rozumné – minimálně ne dostatečně zdůvodněné.

Zejména nerozumím použití databáze, která se udržuje jen v paměti, je k ní program pro její „snadnou úpravu“ (proč databázi neupravuje přímo server a proč by ji někdo chtěl upravovat ručně?) a po každém ukončení serveru se tak její obsah ztratí. Vzhledem k jednoduchosti a velmi malému množství dat by bohatě stačila jednoduchá Java kolekce, i s případnou serializací na disk pro zajištění persistence. Navíc, pokud to chápu správně, program pro obsluhu databáze může dělat cokoli bez jakékoliv autorizace, což jistě není nejlepší řešení. Co zabrání komukoli, aby databázi upravil?

Argumentace pro použití umělé inteligence přímo v klientovi (tedy že bez přístupu na internet by si uživatel nemohl zahrát, přičemž autorka později jako modelový příklad uvádí spuštění serveru na localhost) nedává velký smysl, i když samotné řešení není nutně špatné (je určitě jednodušší než při nutnosti pouštět si server i při hře jednoho hráče).

Návrh a popis komunikačního protokolu je sice poměrně detailní, nicméně z textu vyplývá, že jednotlivé JPanelly implementují vlastní protokol, což jistě není vhodný přístup. Komunikační protokol je celkově velmi nedomyšlený – pokud se například přeruší spojení se serverem, nebo nastane nějaká chyba na serveru, aplikace klienta se ukončí bez jakéhokoliv varování nebo upozornění nebo „zamrzne“. Se serverem klient komunikuje pomocí několika vláken (proč?), a jsou zde omezení typu "nickname hráče nesmí být GAME, protože tento řetězec se používá při komunikaci jako příkaz". Komunikace by měla probíhat pomocí rozesílání herního stavu (který není nijak velký), aby se předcházelo nekonzistencím a komunikace se tak zjednodušila.

Popis implementace v kapitole 3 je sice poměrně podrobný, ale spíše se zaměřuje na informace, které by měly být přítomny v komentářích/generované dokumentaci, kdežto celkový pohled na fungování programu je těžké z textu vyčíst. Příliš tomu nepomáhají ani obrázky (například obr. 3.2).

Z textu mi není jasné, co znamená „skupina“ a kde se do skupiny dá přihlásit. Může server spravovat více skupin? Je pro každou hru vytvořena jedna skupina?

Odkazy na literaturu nejsou v nejvhodnějším formátu, to ale považuji za okrajový detail.

Co je to skupina (na serveru)? Když uživatel při přihlášení zadává jen jméno a heslo, jak se identifikuje skupina, do které se připojil?

Na str. 25 se vyskytuje pojem „nehrající hráč“ – co to znamená?

Proč se po každém tahu kontroluje celé herní pole? Proč se nekontrolují jen části, které se změnilo?

Celkově kvalita textu podle mého nedosahuje úrovně bakalářské práce na MFF UK.

## Implementační část práce

lepší    OK    horší    nevyhovuje

Kvalita návrhu	... architektura, struktury a algoritmy, použité technologie			X	X
Kvalita zpracování	... jmenné konvence, formátování, komentáře, testování			X	X
Stabilita implementace				X	X

Kvalita zpracování implementace je na nízké úrovni, počínaje celkovým návrhem a řadou technických aspektů konče. Celkově působí spíše jako průměrný zápočtový program, ale i tam bych očekával více – ne rozsahem ale kvalitou návrhu a řešení. Uživatelské rozhraní působí divně, na tlačítkách není vidět celý text, okno klienta je třeba po spuštění roztáhnout, aby byly vidět všechny části, jsou použity nestandardní fonty. Celkově je použití aplikace krkolomné – například na konci hry proti počítačovému protivníkovi má uživatel jen možnost zavřít okno, nelze hrát novou hru.

Uvádím další problémy programové části:

- Aplikaci nelze zavřít před koncem hry, „aby to nebylo ostatním hráčům nepříjemné“. To spíše ukazuje na špatný návrh komunikace mezi klientem a serverem. Co když uživatel proces s klientem prostě ukončí?
- Soubor se seznamem agentů (počítačových protivníků) je textový soubor – proč se nepoužije Java reflection nebo *ServiceLoader*, které se pro tento účel přesně hodí?
- Při pokusu o připojení ke spuštěnému serveru tento vypíše „Exception in thread "Thread-0"org.hibernate.HibernateException: hibernate.cfg.xml not found“, i když daný soubor v aktuálním adresáři existuje. Klient se následně zablokuje a nelze ho ukončit jinak než ručním ukončením procesu.

**Celkové hodnocení**    Neprospěl(a)  
**Práci navrhuji na zvláštní ocenění**    Ne

Datum: 23. 8. 2022

Podpis