

Katedra informatiky
Přírodovědecká fakulta
Univerzita Palackého v Olomouci

BAKALÁŘSKÁ PRÁCE

Počítačová hra – Paziaatus



2023

Vedoucí práce:
Mgr. Tomáš Urbanec

Jakub Švihel

Studijní program: Informatika,
Specializace: Programování a vývoj
software

Bibliografické údaje

Autor: Jakub Švihel
Název práce: Počítačová hra – Paziaatus
Typ práce: bakalářská práce
Pracoviště: Katedra informatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci
Rok obhajoby: 2023
Studijní program: Informatika, Specializace: Programování a vývoj software
Vedoucí práce: Mgr. Tomáš Urbanec
Počet stran: 34
Jazyk práce: český

Bibliographic info

Author: Jakub Švihel
Title: Computer game – Paziaatus
Thesis type: bachelor thesis
Department: Department of Computer Science, Faculty of Science, Palacký University Olomouc
Year of defense: 2023
Study program: Computer Science, Specialization: Programming and Software Development
Supervisor: Mgr. Tomáš Urbanec
Page count: 34
Thesis language: Czech

Anotace

Práce pojednává o vývoji desktopové hry Paziaatus, vycházející z karetní hry Pazaak z vesmíru Hvězdných válek a z prohlížečové hry Gladius. Hra je implementována programovacím jazykem Java, vizuální prvky hry jsou vytvořeny pomocí modelu DALL-E 2 a grafického editoru Photopea. Grafické uživatelské rozhraní je vytvořeno vývojovou platformou JavaFX. V práci je popsán vývoj stěžejních mechanismů, tvorba vizuálních prvků a samotný obsah, který hra nabízí

Synopsis

The thesis discusses the development of the desktop game Paziaatus, based on the card game Pazaak from the Star Wars universe and the browser game Gladius. The game is implemented in the Java programming language, the visual elements of the game are created using the DALL-E 2 model and the Photopea graphic editor. The graphical user interface is created by the JavaFX development platform. The work describes the development of key mechanisms, the creation of visual elements and the content itself that the game offers.

Klíčová slova: Pazaak, Hvězdné války, Hra na hrdiny, Java, DALLE-E 2

Keywords: Pazaak, Star Wars, Role-playing game, Java, DALLE-E 2

Děkuji Mgr. Tomáši Urbanci za vedení mé bakalářské práce. Děkuji Kateřině Konečné za dobrovolné testování hry Piziaatus.

Místopřísežně prohlašuji, že jsem celou práci včetně příloh vypracoval samostatně a za použití pouze zdrojů citovaných v textu práce a uvedených v seznamu literatury.

datum odevzdání práce

podpis autora

Obsah

1	Úvod	7
2	Inspirace pro bakalářskou práci	8
2.1	Gladius	8
2.2	Pazaak	9
3	Grafický design a tvorba vizuálních prvků hry	10
3.1	Problém s licencemi	10
3.2	DALL-E 2	10
3.3	Vytváření obrázků předmětů	11
3.4	Úprava vytvořených obrázků předmětů	11
3.5	Uživatelské rozhraní hry	12
3.6	Změna rozlišení hry	13
4	Uživatelská příručka	15
4.1	Ovládání	15
4.2	Hlavní menu	15
4.3	Příběh	17
4.4	Kredity	17
4.5	Karetní hra Paziak	17
4.6	Životní potřeby hráče	19
4.7	Navigační lišta	19
4.8	Inventář	19
4.9	Vybavení	20
4.10	Karetní balíček	20
4.11	Předměty	20
4.12	Vlastnosti předmětů	21
4.13	Práce	22
4.14	Úmrtí hráče	25
5	Programátorská příručka	26
	Závěr	31
	Conclusions	32
A	Obsah přiloženého adresáře	33
	Reference	34

Seznam obrázků

1	Snímek ze hry Gladius	8
2	Snímek ze hry Pazaak	9
3	Vyhovující obrázek bot	11
4	Nevyhovující obrázek bot	11
5	Ukázka upravených předmětů	12
6	Diagram kontejnerových prvků uživatelského rozhraní	13
7	Hlavní menu hry	15
8	Karetní partie hry Paziak	18
9	Vlastnictví hráče	19
10	Mapa lokací	23
11	Lokace migrace	24
12	Diagram tříd	26

1 Úvod

Již od útlého věku se rád věnuji hraní počítačových her. Je to jedna z mých nejoblíbenějších zálib. Rozhodl jsem se můj vztah k hrám prohloubit a to naprogramováním počítačové hry, jakožto závěrečné práce bakalářského studia. Hru jsem zatím nikdy neprogramoval a tak jsem si říkal, že to bude obohacující výzva.

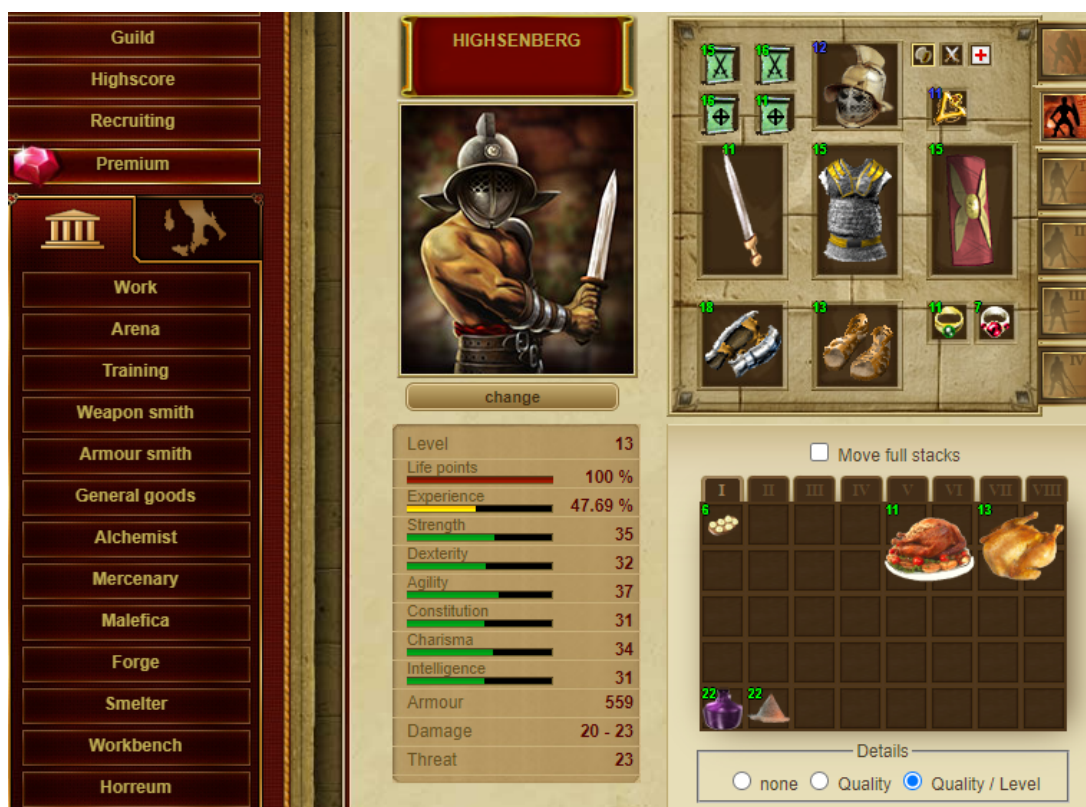
Vytvořil jsem desktopovou hru, která je založena na konceptu již existujících her, avšak s originálním provedením. V rámci práce jsem se zabýval nejen implementací samotné hry, ale také analýzou herních technologií a metodiky vývoje. Hra implementuje algoritmus počítačového protihráče, který je schopný analyzovat situaci a vytvořit strategii během karetní partie. Herní logika je napsána v programovacím jazyce Java. Vizuální prvky jsem vytvořil pomocí modelu DALLE-E 2 a webového editoru Photopea.

2 Inspirace pro bakalářskou práci

V době, kdy jsem vymýšlel téma bakalářské práce, jsem aktivně hrál počítačovou hru Hvězdné války: Rytíři Staré Republiky, jejíž součástí je karetní hra Pazaak a prohlížečovou hru Gladius. Napadlo mě, že spojením karetní hry Pazaak a hry Gladius by mohla vzniknout zajímavá hra kombinující prvky videoherního žánru hra na hrdiny s hazardními prvky. Sloučením názvů Pazaak a Gladius vznikl název Paziaatus pro mou bakalářskou – práci.

2.1 Gladius

Jedná se o online prohlížečovou hru s prvky RPG odehrávající se v antickém Římě. Hráč hraje roli gladiátora, který může bojovat proti různým nepřátelům nebo chodit do práce. Skrz tyto aktivity vydělává zlaté mince a rubíny, za které lze kupovat předměty, kterými jsou například zbraně, oblečení a jídlo. Gladius obsahuje různé lokace, kde každá má svou specifickou funkci. Hráči spolu mohou komunikovat přes chat, bojovat proti sobě v arénách, či seskupovat se v guildách, pomocí kterých mohou chodit na společné výpravy. Hra se ovládá převážně pomocí myši. [1]



Obrázek 1: Snímek ze hry Gladius

2.2 Pazaak

Tato karetní hra z vesmíru Hvězdných válek je velmi podobná karetní hře Blackjack. Cílem hry je získat více bodů než soupeř a zároveň hodnota všech vyložených karet nesmí překročit hodnotu 20. Hráči mají možnost použít speciální karty, kterými lze změnit celkovou hodnotu vyložených karet. Výherce získává smluvený počet kreditů, za které lze koupit další speciální karty. Kredity jsou herní měna. Výhru karetní partie ovlivňuje náhoda společně se zvolenou strategií. Hra se ovládá pouze pomocí myši. [2]



Obrázek 2: Snímek ze hry Pazaak

3 Grafický design a tvorba vizuálních prvků hry

Herní grafika je klíčovým prvkem, který ovlivňuje zážitek hráče a přispívá k celkovému dojmu a atmosféře hry. Z toho důvodu jsem si na tvorbě vizuálních prvků dal velmi záležet. Vývoj hry Paziaatus jsem započal tvorbou vizuálních prvků, které jsem věnoval přibližně stejné množství času jako vymýšlení logiky samotné hry. Umělecký styl hry Paziaatus je futuristický.

3.1 Problém s licencemi

Během prvotních návrhů hry jsem přemýšlel, jakým způsobem získat, nebo vytvořit obrázky předmětů, jako jsou světelné meče, zbroje, vznášedla a tak dále. Jako nejjednodušší řešení mě napadlo využít obrázky, které již někdo vytvořil. Zde ale nastal problém s licencemi. Zkontaktoval jsem společnosti LucasArts a Electronic Arts, kteří jsou vlastníci licencí obrázků z her s tematikou hvězdných válek, ale sdělili mi, že jejich obrázky nesmím jakkoliv využívat. Dále jsem zkontaktoval 6 digitálních tvůrců, ale svolení mi dal pouze jeden. Svolil mi využívat obrázky herních karet, které jsem potřeboval pro karetní část hry. [3]

S dalšími obrázky předmětů, které jsem na internetu našel, jsem byl nespokojený, nebo měly placené licence. Rozhodl jsem se tedy zbytek obrázků vytvořit. Na internetových fórech jsem se dočetl o DALL-E 2 a rozhodl jsem se ho použít pro tvorbu obrázků.

Názvy z vesmíru Hvězdných válek jsou také licencovány. Proto se názvy ve hře Paziaatus liší od licencovaných názvů. Například místo hry Pazaak jsem karetní hru pojmenoval Paziak a podobně.

3.2 DALL-E 2

Moderní generativní model DALL-E 2 [4], vyvinutý společností OpenAI, představuje významný pokrok mající potenciál způsobit revoluci, jakým vytváříme digitální obsah. Je schopen vytvářet obrázky s vysokou kvalitou a rozlišením 1024×1024 pixelů. Využívá více než 10 miliard různých parametrů, což mu umožňuje učit se složitější vzory a vytvářet rozmanité a realistické obrázky. Dokáže vytvářet obrázky s širokou škálou objektů a scén, včetně předmětů, zvířat, lidí a dokonce i abstraktních konceptů. Čím podrobněji uživatel popíše vzhled obrázku, tím přesnější bude výsledek. Ve výsledném obrázku je v pravém dolním rohu umístěn vodotisk, společnost OpenAI však povoluje jeho odstranění.

Pro vytváření obrázků je využíván algoritmus hlubokého učení z textových popisů. DALL-E 2 nejprve zpracuje vstupní text pomocí transformátorových sítí, které mu umožňují porozumět významu a kontextu textu. Poté pomocí konvoluční neuronové sítě převede text na obrázek tím, že identifikuje a kombinuje různé vizuální prvky, jako jsou tvary, barvy a textury.

3.3 Vytváření obrázků předmětů

Při vytváření obrázků předmětů jsem v textovém vstupu, který jsem předal modelu DALL-E 2, co nejpřesněji specifikoval, jaký konkrétní předmět chci vytvořit. Zejména jaký tematický vzhled by předmět měl mít, které barvy by měl obsahovat a pomocí jakého stylu by měl být vytvořen. Například: pár vysokých bot s futuristickou tematikou, který má tmavě červenou a tmavě oranžovou barvu a je vytvořen ve vektorovém stylu.

Ačkoliv je DALL-E 2 velmi pokročilý model a je schopen vytvořit vyhovující a použitelné obrázky, tak pro totožný textový vstup dokáže vytvořit obrázky, které použitelné nejsou. Z 673 vytvořených obrázků předmětů, bylo použitelných přibližně 210 obrázků.



Obrázek 3: Vyhovující obrázek bot



Obrázek 4: Nevhovující obrázek bot

3.4 Úprava vytvořených obrázků předmětů

Vytvořené obrázky byly potřeba dále upravit. Pro úpravu jsem zvolil webový editor Photopea [5], který umožňuje vytvářet, upravovat a manipulovat s obrázky. Jedná se o bezplatný editor, nabízející mnoho funkcí, které lze nalézt v placených profesionálních editorech pro úpravu obrázků.

Při úpravě obrázků předmětů jsem vždy postupoval tak, že jsem odstranil vodotisk, upravil jsem barevné pozadí na transparentní, přidal černý obrys, vycentroval předmět, upravil rozlišení na hodnotu 100×100 pixelů a případně poupravil barvy. Výsledný obrázek předmětu jsem převedl do formátu PNG, jelikož podporuje průhledná pozadí. Hra Paziiaatus obsahuje 196 obrázků předmětů.



Obrázek 5: Ukázka upravených předmětů

3.5 Uživatelské rozhraní hry

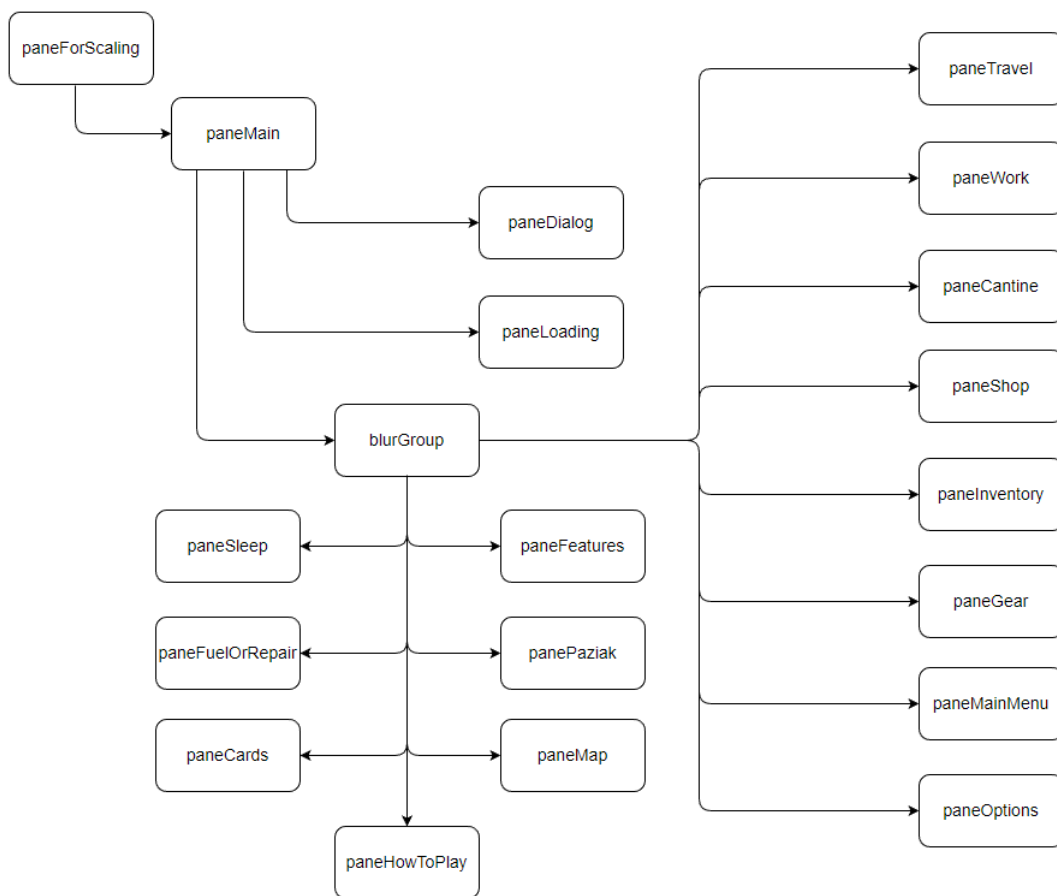
Komponenty uživatelského rozhraní jsem vytvořil pomocí editoru Photopea. Inspiroval jsem se komponentami z webové stránky freepik.com. Při tvorbě jsem se snažil, aby byly komponenty co nejvíce intuitivní a atraktivní. Barvu komponent jsem zvolil pouze černou, bílou a oranžovou, aby byly dobře čitelné a co nejjednodušší na vytvoření. Hře jsem vytvořil modré, červené a zelené pozadí, na které jsem aplikoval styl překrytí vzorem. Použil jsem vzor z webových stránek brusheezy.com [6], ke kterým se vztahuje komunitní licence Brusheezy, která povoluje vzor použít pro osobní i komerční účely.

Rodinu písma veškerého textu hry jsem zvolil Sofachrome [7], protože její vzhled je futuristický, podporuje diakritická znaménka písmen a je bezplatný pro osobní i komerční použití.

Pro vizualizaci lokací a některých tlačítek jsem využil ikony z webové stránky game-icons.net [8], jelikož jsou poskytovány pod licencí CC BY 3.0, díky které je možné obsah použít pro osobní i komerční účely. Licence vyžaduje, abych informoval, že jsem provedl změny některých použitých ikon.

Vyhotovené komponenty, mezi které patří například tlačítka, čítače bodů, prapory obchodů a tak dále, jsem poskládal v ucelené uživatelské rozhraní vizuálním nástrojem Scene Builder, který je určený pro vytváření JavaFX aplikací. Umožňuje vytvářet a upravovat uživatelské rozhraní aplikace operacemi táhni a pusť. JavaFX je open-source vývojová platforma vyvinutá společností Oracle Corporation [9].

Kořenovým kontejnerovým prvkem uživatelského rozhraní je panel paneForScaling, jehož potomkem je panel paneMain. Funkce panelu paneMain je načtení kaskádových stylů. Potomky paneMain jsou kontejnerové prvky, kterými jsou panely paneLoading a paneDialog a skupina blurGroup. Skupina blurGroup obsahuje většinu prvků uživatelského rozhraní, tedy například panely paneInventory, panePaziak, paneMainMenu a mnoho dalších. Tyto panely obsahují konkrétní komponenty uživatelského rozhraní. Význam panelů paneDialog a paneLoading je rozmazat a deaktivovat veškeré prvky skupiny blurGroup a zobrazit načítací, nebo dialogové okno.



Obrázek 6: Diagram kontejnerových prvků uživatelského rozhraní

3.6 Změna rozlišení hry

Možnost měnit velikost herního okna pomocí posuvníků je zakázána, kvůli možné degeneraci vzhledu komponent uživatelského rozhraní a obrázků předmětů. Bylo ale nutné vyřešit problém s rozlišením hry, aby byla zajištěna kompatibilita s různými monitory, které nemusí podporovat dané rozlišení. Chtěl jsem, aby bylo možné si vybrat ze sedmi různých rozlišení.

Jediná možnost, která mě napadla, byla vytvořit všechny komponenty uživatelského rozhraní a obrázky předmětů v sedmi různých rozlišeních a podle toho, jaké rozlišení by hráč zvolil, tak by se změnil rozměr herního okna a vybraly komponenty a obrázky s požadovaným rozlišením. To by ale bylo neefektivní. Hra by musela obsahovat sedm obrázků pro každý předmět. Byl zapotřebí soubor, který by uchovával údaje o souřadnicích, kde se má objekt zobrazovat pro zvolené rozlišení, pro každý objekt. Při každém spuštění hry nebo změně rozlišení, by se musely nastavit souřadnice všech objektů, což by mělo za důsledek nechtěnou režii navíc. Rozhodl jsem se problém změny rozlišení nechat na závěrečnou fázi vývoje aplikace.

Skrze používání vizuálního nástroje Scene Builder jsem objevil vlastnost scale.

Tato vlastnost udává v jakém měřítku se má prvek zmenšit, nebo zvětšit. Vytvořil jsem panel `paneForScaling` a nastavil ho jako kořenový panel. Funkce tohoto panelu je zobrazit uživatelské rozhraní ve zvoleném rozlišení skrze změnu vlastnosti `scale`. Funkcionalita změny rozlišení je podrobněji popsána v části programátorská příručka.

Hra *Paziaatus* podporuje rozlišení 1600×900 , 1440×810 , 1280×720 , 1120×630 , 960×540 , 800×450 a 640×360 .

4 Uživatelská příručka

Část podrobně popisující hru.

4.1 Ovládání

Hra se ovládá pouze levým a pravým tlačítkem myši. Herní tlačítka reagují pouze na levé tlačítko myši.

4.2 Hlavní menu

Při startu hry se načte hlavní menu.

Ozubené kolečko – Kliknutím se otevře panel s nastavením rozlišení hry.

Postava s otazníkem – Kliknutím se otevře panel s vysvětlením, co jednotlivé vlastnosti předmětů ovlivňují.

Myš s otazníkem – Kliknutím se otevře panel s vysvětlením, jak se hra ovládá.

Tlačítko UKONČIT – Kliknutím se ukončí hra.



Obrázek 7: Hlavní menu hry

Herní pozice

Nová hra – Kliknutím na tlačítko nová hra se otevře okno, kde je třeba vybrat herní mód a specializaci postavy. Po výběru stačí kliknout na tlačítko ANO a

spustí se nová hra.

Načíst hru – Kliknutím na tlačítko, kde je uvedený čas a datum se načte vybraná uložená herní pozice a můžete pokračovat s hraním hry Paziaatus tam, kde jste skončili.

Koš – Kliknutím na tlačítko koš se otevře dialogové okno, s možností smazat danou herní pozici. Kliknutím na tlačítko NE se uložená pozice nevymaže. Kliknutím na tlačítko ANO se uložená pozice vymaže.

Herní módy

Volba herního módu značně ovlivní herní zážitek. Ovlivňuje strategii hráče a rychlost průchodu hrou. Volba herního módu probíhá při vytvoření nové hry a volbu v pozdější fázi hry nelze změnit.

Klasický mód - Hra běží ve standardním čase. Po úmrtí je možné načíst poslední uloženou pozici.

Rychlý mód - Hra běží se zrychleným časem. Doba trvání misí a cestování je dvakrát kratší. Pokles životních potřeb je zrychlen o jednu čtvrtinu. Doba, po které dojde k devastaci planety Tarrys je kratší o jednu čtvrtinu. Po úmrtí je možné načíst poslední uloženou pozici.

Realistický mód - Hra běží ve standardním čase. Po úmrtí není možné načíst poslední uloženou pozici, tedy uložená pozice je odstraněna a je nutné začít novou hru.

Specializace postavy

Obtížnost průchodu hrou ovlivňuje volba specializace postavy, mezi které patří civilista, lovec odměn a šedý válečník. Volba specializace probíhá při vytvoření nové hry.

Specializace civilista je těžká obtížnost. Nemá žádné bonusy při užívání specifických zbraní. Nemůže se vybavit artefaktem ani implantátem.

Specializace lovec odměn je střední obtížnost. Má malý bonus k útoku při užívání střelných zbraní. Může své vybavení obohatit o implantát.

Specializace šedý válečník je lehká obtížnost. Má střední bonus k útoku při užívání kontaktních zbraní. Může se vybavit implantátem i artefaktem.

4.3 Příběh

Během cesty z práce do kantýny hráč zaslechne rozhovor dvou poručíků Telosianského impéria, jak se baví o útoku na planetu Tarrys, na které se nachází. Brzy má zaútočit Telosianský křižník a zdevastovat tak celou planetu. Úkolem hráče je vydělat dostatečné množství kreditů a zakoupit letenku na nejbližší planetu Narr Sheyda, aby se zachránil. Jestliže hráč nestihne odcestovat do 15 dní, tak je během ničení planety zasypán sutinami vysokých domů a hra pro něj končí. 15 dní odpovídá jedné hodině.

Při příletu na Narr Sheyda si cestující odchytne gang Seroko, který po všech vymáhá kredity, jelikož přistáli na přistávací ploše v sektoru ovládaném tímto gangem. Protože hráč nemá dostatečné množství kreditů na zaplacení, je zbit a okraden o většinu výbavy. Nezaplacením si hráč znepřátelil gang Seroko a tak je jeho činnost na této planetě občas narušena přepadením tímto gangem. Jediným východiskem, jak zabránit těmto neustálým přepadením, je odcestovat na planetu Kerasant.

4.4 Kredity

Kredity slouží ve hře Paziaatus jako platidlo. Jsou nutné k pronájmu noclehu, migraci na ostatní planety a nakupování jídla, pití, karet, vybavení a tak dále. Kredity je možné získat prodejem získaného vybavení, plněním misí nebo vyhráním karetní partie.

4.5 Karetní hra Paziak

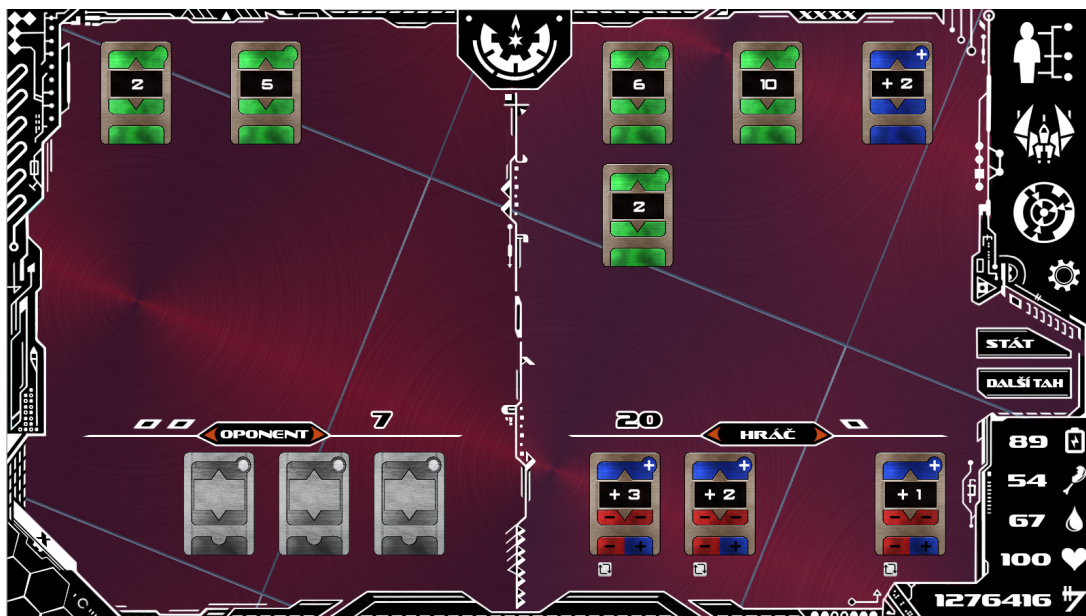
Paziak je karetní hra inspirovaná karetní hrou Pazaak. Hra se hraje se zamíchaným hlavním balíčkem obsahující 40 karet číslovaných od 1 do 10 s kladnými hodnotami a balíčkem hráče, který obsahuje 10 speciálních karet. Na začátku karetní partie se každému hráči vytvoří vedlejší balíček ze 4 náhodně vybraných speciálních karet z balíčku hráče a přidělí 1 karta z hlavního balíčku. Přidělené a použité karty se vykládají na hrací stůl. Nazývají se vyložené karty. Hráč má možnost během svého tahu jednu z karet vedlejšího balíčku použít a změnit tak celkovou hodnotu svých vyložených karet.

Hráči, který nestojí, je na začátku jeho tahu přidělena karta z hlavního balíčku, jehož hodnota se sčítá s ostatními vyloženými kartami. Na konci tahu se hráč rozhoduje, zda-li ukončí tah, nebo bude stát. Ukončení tahu znamená, že hráč chce, aby mu byla přidělena v následujícím kole další karta. Rozhodne-li se hráč stát, tak mu nebude přidělena další karta z hlavního balíčku a čeká, až bude stát i protivník. Když oba hráči stojí, tak dochází k vyhodnocení a určení výherce setu.

Set vyhrává ten, kdo má celkovou hodnotu vyložených karet nejbližší 20, aniž by tato hodnota byla překročena. Pakliže mají oba hráči stejnou hodnotu vyložených karet, případně mají oba hráči hodnotu vyšší než 20, tak nastává

remíza. Kdo první získá 3 sety, tak vyhrál karetní partii a obdrží od protihráče domluvené množství kreditů.

Paziak nelze hrát, pokud hráčův balíček neobsahuje 10 speciálních karet. Speciální karty lze zakoupit v kantýně.



Obrázek 8: Karetní partie hry Paziak

Typy speciálních karet

Karty +1 až +6, -1 až -6 a ± 1 až ± 6 upravují celkovou hodnotu vyložených karet o částku zobrazenou na kartě. U karet se znaménkem \pm si hráč vybere, zdali se má hodnota karty odečíst, či přičíst k celkové hodnotě vyložených karet.

Karta 1 ± 2 může změnit své znaménko i hodnotu. To znamená, že karta může nabývat hodnot +1, -1, +2, nebo -2.

Karta 3&6 změní znaménko všech vyložených karet s hodnotou 3 a 6 a stejně tak karta 2&4 změní znaménko všech vyložených karet s hodnotou 2 a 4.

Příklad: Hodnota vyložených karet je $10 + 7 + 6 = 23$. Nyní, když se použije karta 3&6, se znaménko vyložených karet s hodnotou 3 a 6 změní. Tedy hodnota vyložených karet je $10 + 7 - 6 = 11$.

Karta D (Double) zdvojnásobuje hodnotu poslední vyložené karty. Lze to také chápat tak, že karta Double má stejnou hodnotu jako právě vyložená karta.

Karta $\pm 1T$ (Tie breaker) je speciální ± 1 karta. Pakliže by měla nastat mezi hráči remíza, ale některý z hráčů použil tuto kartu, tak remíza nenastává a set

vyhrává hráč, který tuto kartu použil. Aby tato karta měla efekt rušící remízu, tak musí být použita jako poslední karta v setu. Není-li použita jako poslední karta setu, je brána v potaz pouze hodnota +1, či -1.

4.6 Životní potřeby hráče

Mezi životní potřeby hráče se řadí sytost, hydratace a energie. Hráč pocituje hlad, žízeň a únavu, což znamená, že každých 30 vteřin se sníží hodnoty životních potřeb. Při klesnutí pod hranici 20 % se postavě začíná zvyšujícím tempem snižovat zdraví. Je nutné postavě kupovat jídlo a pití, aby byl sytý a hydratovaný. Taktéž je potřeba pronajímat si nocleh, aby byla hráči doplněna energie. Ztracené zdraví lze doplnit užitím různých medikamentů.

4.7 Navigační lišta

V pravém horním rohu hry se nachází navigační lišta. Po kliknutí na ikonu postavy se zobrazí vše, co hráč vlastní. Druhá ikona znázorňuje lokaci, ve které se hráč aktuálně nachází. Třetí ikona znázorňuje mapu, skrz kterou může hráč cestovat do různých lokací. Ikona dveří slouží k opuštění hry.



Obrázek 9: Vlastnictví hráče

4.8 Inventář

Inventář slouží k uchování zakoupených či získaných předmětů.

4.9 Vybavení

Vybavení jsou předměty, které postava hráče aktivně používá. Jinak řečeno, hráč je předmětem vybaven. Pro vybavení se předmětem z inventáře je nutné kliknout na předmět pravým tlačítkem. Aktivním používáním se předměty opotřebovávají a zhoršuje se tak jejich stav.

4.10 Karetní balíček

Hráč si může svůj herní balíček obohacovat zakoupením nových speciálních karet v kantýně. Herní balíček může obsahovat pouze speciální karty.

4.11 Předměty

Oblečení

Do kategorie oblečení patří předměty typu rukavice, helmy, boty, opasky a oděvy. Zvyšují obranu postavy. Oblečení je středně těžké.

Zbraně

Mezi zbraně patří předměty typu světelné meče a laserové pušky. Tyto předměty jako jediné zvyšují útok postavy. Zbraně jsou středně těžké.

Nástroje

Nástroje jako jediné zvyšují zručnost postavy. Nástroje jsou velmi těžké.

Artefakty a implantáty

Artefakty a implantáty jsou předměty, které lze získat pouze prostřednictvím plnění archeologických a technických misí, tedy není možné je koupit v obchodě. Téměř nic neváží

Prsteny a náhrdelníky

Předměty zvyšující hlavně vzhled a štěstí postavy. Téměř nic neváží.

Karty

Karty jsou nutné k vytvoření karetního balíčku. Pokud karetní balíček hráče neobsahuje přesně 10 speciálních karet, tak nemůže hrát karetní hru Paziak.

Jídlo a pití

Zvyšují sytost a hydrataci hráče a v malém množství i zdraví hráče. Při použití se předmět spotřebuje.

Medikamenty

Medikamenty obnovují zdraví hráče, ale snižují hráčovu sytost a hydrataci. Při použití se předmět spotřebuje.

4.12 Vlastnosti předmětů

Vlastnosti předmětů, které se nacházejí v obchodě, se při naskladnění nového zboží vždy mění, a proto je téměř nemožné, narazit v obchodě na dva totožné předměty.

Útok

Vyšší vlastnost znamená větší udělené poškození v souboji. Velmi důležitá vlastnost pro práci lovec. Středně důležitá pro práce archeolog a technik. Nepodstatná pro práci pracovník a pro karetní hru Paziak.

Obrana

Obrana je důležitá vlastnost pro práci lovec. Středně důležitá pro práce archeolog a technik. Nepodstatná pro práci pracovník a pro karetní hru Paziak. Vyšší vlastnost znamená větší odolnost vůči poškození v souboji.

Zručnost

Zručnost ovlivňuje rychlost provedení práce jako pracovník. Nepodstatná pro práci lovec, archeolog a technik a pro karetní hru Paziak.

Štěstí

Důležitá vlastnost pro karetní hru Paziak. Nepodstatná pro jakoukoli práci. Vyšší vlastnost znamená větší šance přidělení karty s nižší hodnotou ve hře Paziak.

Vzhled

Vyšší vzhled znamená možnost vsadit větší částku kreditů v karetní hře Paziak. Jde o důležitou vlastnost pro karetní hru Paziak. Vlastnost je nepodstatná pro jakoukoli práci.

Váha

Čím vyšší váha, tím delší doba pro vyhotovení práce a delší trvání cestování po různých lokacích. Všeobecně důležitá vlastnost.

Rychlost

Vlastnost ovlivňuje dobu cestování z práce a do práce. Ovlivňuje trvání cestování po různých lokacích. Všeobecně důležitá vlastnost.

Palivo

Říká kolik zbývá vznášedlu paliva. Pokud hodnota klesne na nulu, tak dochází k velmi výraznému zpomalení hráče.

Cena

Udává částku, za kolik může hráč prodat daný předmět. Při nákupu předmětu se cena předmětu sníží.

Stav

Říká jak moc je předmět opotřebený. Pokud hodnota klesne na nulu, tak je předmět zničen.

Možné opravy

Udává, maximálně kolikrát je možné předmět opravit.

4.13 Práce

Je možné vykonávat práci jako lovec, archeolog, technik nebo pracovník. Práce lovce má vysokou šanci, že hráč během práce zemře, ale má nejvyšší výdělek. Práce archeologa nebo technika má střední šanci, že hráč během práce zemře a mají střední výdělek, ale zároveň je šance že během práce najdou artefakt, nebo vyrobí implantát, který si hráči mohou nechat. Práce jako pracovník je nejméně výdělečná, ale má nejmenší šanci, že hráč zemře.

Droidi

Pokud hráč vlastní droida, tak ho může poslat do práce místo sebe a neriskuje tak vlastní smrt. Když je droid v práci, tak hráč mezitím může například nakupovat vybavení nebo hrát karetní hru Paziak.

Vznášedla

Rapidně zrychlují cestování mezi lokacemi a transport z práce a do práce. Musí se jim doplňovat palivo. Když se hráč vydá do práce bez paliva, je jeho cestování pomalejší, než kdyby šel práce bez vznášedla.

Lokace

Po kliknutí na ikonu mapy v navigační liště se zobrazí lokace, na které může hráč cestovat. Cestování do lokací trvá určitou dobu, protože se do dané lokace musí hráč přesunout. Čím vyšší je zátěž hráče, tím déle cestování trvá. Během přesunu mezi lokacemi může být hráč přepaden a nastane tak souboj. Cestování mezi lokacemi se dá zrychlit zakoupením vznášedla.



Obrázek 10: Mapa lokací

Obchody

Lokace, kde hráč může pravým stisknutím tlačítka prodat nebo koupit různé předměty. Levým kliknutím si může hráč zobrazit vlastnosti zvoleného předmětu. Ve hře se nacházejí obchody se zbraněmi, zbrojí, medikamenty, potravinami, šperky a technikou.

Kantýna

Lokace určená k nákupu speciálních karet. Také se zde odehrávají partie karetní hry Paziak.

Nocleh

V této lokaci si hráč může vybrat ze dvou typů noclehů. První nocleh je levnější, ale nedoplňuje hráči tolik energie a je zde šance, že je hráč během spánku přepaden. Druhý nocleh je sice dražší, ale doplní hráči větší množství energie a není zde šance, že by byl hráč během spánku přepaden.

Pronajmutím noclehu a vyspáním se si hráč ukládá svou herní pozici. Po vyspání všichni obchodníci doplní zboží.

Čerpací stanice

Lokace určená k natankování vznášedla. Cena paliva se mění podle toho, na jaké planetě se hráč aktuálně nachází.

Opravář

Lokace určená k opravě opotřebeného předmětu. Opravením se sníží hodnota vlastnosti možné opravy.

Migrace

Migrace znamená přesun postavy na jinou planetu. Postup ve hře se dá měřit podle toho, na jaké planetě se hráč aktuálně nachází. Planeta, na které hráč začíná novou hru, se jmenuje Tarrys. Následující planeta se jmenuje Narr Sheyda a konečná planeta je Kerasant.

Postupnou migrací na další planety si hráč odemyká nové možnosti, mezi které patří například koupě nebo získání nových předmětů, vyšší pracovní výnosy, odemčení dalších lokací, navýšení slotů obchodů a podobně. Dále se mírně zvyšuje obtížnost protihráče během karetní partie, protože protihráč využívá lepší speciální karty. K migraci na jiné planety je potřeba vysokého obnosu peněz. Migrací si hráč odemyká možnosti přepínat mezi barevnými pozadími hry.



Obrázek 11: Lokace migrace

4.14 Úmrtí hráče

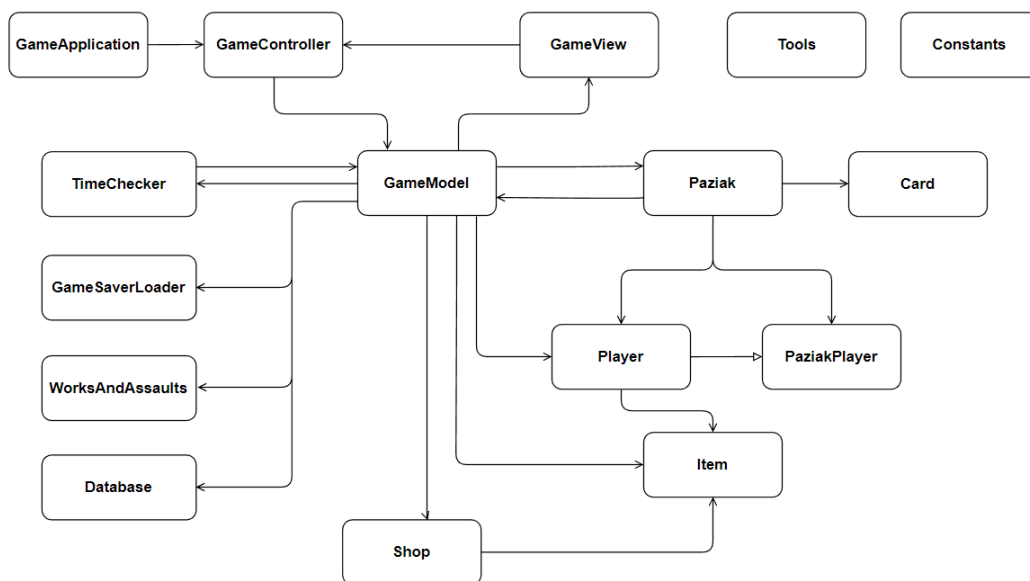
Hráč umírá, pakliže jeho zdraví klesne na hodnotu 0. Zdraví snižuje hladovění, žíznění a energetické vyčerpání hráče. Zdraví také snižují souboje.

Jestliže si hráč zvolil realistický herní mód, tak je herní pozice přemazána a musí si založit novou hru. V ostatních případech může hráč načíst poslední uloženou pozici.

5 Programátorská příručka

Logika hry Paziatus je implementována ve vysokoúrovňovém objektově orientovaném programovacím jazyce Java [10]. Použil jsem vývojové prostředí Eclipse. Databázi pro perzistentní uložení základních vlastností předmětů jsem zvolil H2. Jde o lehkou, vysoce výkonnou a zabudovatelnou databázi. Pro připojení a správu databáze jsem zvolil open-source multiplatformní nástroj DBeaver. Pro ukládání a načítání hry je použita open-source knihovna Gson [11]. Pomocí nástroje Maven jsem vytvořil Java Archive (JAR) soubor, který slouží ke spuštění hry.

K vytvoření hry jsem použil vzor model-view-controller a rozdělil tak architekturu do tří hlavních tříd a třinácti vedlejších tříd. Oddělil jsem tak řídicí logiku, datový model a uživatelské rozhraní hry. Programovou část vývoje hry jsem začal s programováním hlavních tříd GameController, GameModel a GameView a poté jsem začal programovat třídy vedlejší. Hra je rozdělena do balíčků main, beeings, paziak a residue. Balíček main obsahuje zmíněné hlavní třídy. Balíček beeings obsahuje třídu PaziakPlayer a třídu Player. Balíček paziak obsahuje třídy, které jsou potřebné pro fungování karetní hry Paziak. Balíček residue obsahuje zbylé třídy hry.



Obrázek 12: Diagram tříd

GameApplication.java

Třída určená pro spuštění hry Paziatus. Obsahuje metodu main. Třída například načítá kořenový prvek uživatelského rozhraní ze souboru game.fxml nebo zakazuje měnit velikost okna pomocí posuvník. Má důležitou roli při zapínání

hry, protože kontroluje, zda-li už neběží jiná instance hry. Kdyby mohlo běžet zároveň více instancí hry, tak by se hra mohla dostat do chybového stavu. Informace o tom, jestli běží jiná instance hry, je čtena a zapisována do souboru `game.properties`.

GameController.java

Třída inicializuje herní stav, spravuje prvky uživatelského rozhraní a zpracovává uživatelské akce, kterými jsou například kliknutí na tlačítka myši a ty předává třídě `GameModel`.

GameModel.java

Třída uchovávající informace o herním stavu a obsahující většinu logiky hry. Komunikuje s třídou `GameView` kvůli vykreslování prvků. Vytváří instanci třídy `Database`, skrz kterou volá metody pro tvorbu předmětů. Komunikuje s třídou `GameSaverLoader`. Skrze komunikaci ukládá a načítá herní obsah. Vytváří vlákno, které má na starost po určitých časových jednotkách informovat o tom, že se mají snížit životní potřeby hráče.

GameView.java

Třída, která se stará o vizuální stránku hry. Řeší nastavování viditelnosti prvků uživatelského rozhraní, přepínání panelů, změnu textu prvků a další.

```
protected void changeResolution()
```

Metoda je volána při změně rozlišení herního okna. Vypočítá se hodnota měřítko, vůči kterému se má herní okno zmenšit, nebo zvětšit. Hodnota se vypočítá podílem zvoleného rozlišení a rozlišením 1600×900 . Tato hodnota se nastaví pomocí metod `setScaleX` a `setScaleY`, což ovlivní velikost všech grafických prvků hry. Dále se všechny prvky uživatelského rozhraní posunou do levého horního rohu. Nakonec se změní výška a šířka herního okna a okno se vycentruje.

```
private void disableTravellingImages(boolean disable)
```

Metoda, která slouží k vyblednutí a vypnutí tlačítek pro cestování po mapě. Tlačítka se opět vybarví a zapnou po dovršení 100 % indikátoru pokroku, který trvá specifický čas podle zvoleného herního módu a zátěže hráče. Tohle čekání reprezentuje reálné cestování mezi lokace.

PaziakPlayer.java

Třída slouží k vytvoření objektu karetního hráče.

Card.java

Objekty této třídy reprezentují karty pro karetní hru Paziak.

```
public boolean isCardUsed()
```

Metoda informuje skrze návratovou hodnotu, zdali byla karta vedlejšího balíčku použita.

```
public void useCard()
```

Metoda nastavuje kartě vedlejšího balíčku informaci, že byla použita.

Player.java

Třída určená pro vytvoření objektu, který představuje hráče hry Paziak. Objekt uchovává informace o předmětech, které jsou součástí hráčova vybavení, inventáře a karetního balíčku. Uchovává hodnoty životních potřeb, životů, vlastností vybavení, počtu kreditů, aktuální planety a zátěže vybavení, inventáře a karetního balíčku. Třída Player.java dědí z třídy PaziakPlayer.java, jelikož hráč hry je zároveň karetní hráč.

```
public int addItemIntoInventory(Item item)
```

Metoda přidá zakoupený předmět, nebo sundané vybavení do inventáře na volný slot. Vrací číslo slotu, do kterého byl předmět uložen. Jestliže je inventář plný, tak předmět není do inventáře přidán a návratová hodnota je rovna hodnotě -1. Maximální kapacita inventáře je dána konstantou PLAYERS_INVENTORY_SIZE třídy Constants, jejíž výchozí hodnota je 12.

Paziak.java

Třída obsahuje logiku karetní hry Paziak a algoritmus karetního protihráče. Uchovává informace o průběhu karetní parite.

```
public void opponentsTurn() throws InterruptedException
```

Metoda znázorňuje tah protihráče. Protihráč se rozhoduje, zda pokračovat dalším tahem, nebo stát. Algoritmu protihráče jsem implementoval strategii, kterou během hraní karetní hry Paziak osobně používám.

Protihráč se rozhodne stát, například pokud má hodnotu vyložených karet 20, pokud má vyšší hodnotu vyložených karet než hráč hry a hráč hry zároveň stojí, pokud má hodnotu vyložených karet 18, nebo 19 a hráč hry nestojí.

Protihráč se rozhodne pokračovat dalším tahem, například pokud má hráč hry hodnotu vyložených karet 20, pokud je hodnota vyložených karet protihráče nižší než 18, pokud hráč hry má vyšší hodnotu vyložených karet než protihráč a zároveň hráč hry nemá vyšší hodnotu vyložených karet než-li 20.

```
public boolean opponentConsidersUsingACard(  
List<SideDeckCard> cards)
```

Argument cards obsahuje použité i nepoužité vedlejší karty oponenta. Protihráč skrze metodu zjišťuje, zda-li se mu vyplatí vyložit některou z jeho karet z vedlejšího balíčku, nebo ne. Pokud se hodnota vyložených karet společně s hodnotou některé vedlejší karty rovná hodnotě 18, 19, nebo 20 a zároveň nebude vyložená vedlejší karta bez účinku, tak protihráč vyhovující kartu vedlejšího balíčku vyloží.

Karta bez účinku znamená, že i přes její vyložení, se nezmění výhra hráče hry. Například pokud protihráč použije kartu, díky které získá hodnotu vyložených karet 19, ale hráč hry má hodnotu vyložených karet 20, tak se nic nezmění a hráč hry vyhrává. V tomto případě protihráč kartu nepoužije a bude si ji šetřit na další set.

```
private void endOfTurn(int valueOfTemporaryUsedCard)  
throws InterruptedException
```

Metoda, která vyhodnocuje stav hry a kontroluje, jestli někdo nezískal výherní set, jestli nenastala remíza a jestli někdo nevyhrál karetní partii.

Constants.java

Třída obsahuje všechny konstanty, které hra využívá. Založení třídy pro konstanty jsem udělal z důvodu větší přehlednosti, z důvodu přístupu ke konstantám z více tříd a také z důvodu, kdyby se někdo rozhodl hru upravit, ale neovládal by programování, jelikož stačí pouhá změna hodnot některých konstant. Třída se dělí na konstanty, které lze upravit, aniž by bylo nutné měnit logiku hry a na konstanty, které vyžadují přepracování logiky hry při jejich změně.

Database.java

Třída určená pro práci s databází. Vytváří předměty obchodů.

```
private void createSpecificShopItems(Shop shop,  
String actualPlanet)
```

Metoda, která z dat z databáze vytvoří předměty a poté předměty vloží do obchodu, který je předán argumentem shop. Je náhodně vybráno, které předměty se budou vytvářet. Vytváří se pouze předměty, které jsou zakoupitelné na planetě, která je předána argumentem actualPlanet. Data z databáze jsou základ pro vlastnosti předmětů. Finální vlastnosti předmětu jsou tvořeny náhodně, aby byl každý předmět odlišný.

GameStateSaverLoader.java

Třída řešící ukládání a načítání herní pozice. Slouží pro čtení a zapisování souboru game.properties.

```
protected void saveGameState()
```

Metoda vytvoří instanci typu Gson, která je použita pro převedení objektu hráče na typ řetězec ve formátu pro soubor .json. Zapisovač writer poté zapíše obsah řetězce do souboru formátu .json.

```
protected void loadGameState(int nthGameState)
```

V metodě čítač reader čte obsah souboru formátu .json a zapisuje ho do řetězce. Vytvoří se instance typu Gson a ta vytvoří objekt hráče pomocí obsahu řetězce.

Item.java

Třída určená pro vytvoření objektu, který představuje herní předmět.

WorksAndAssaults.java

Třída obsahující logiku, která řeší chození do práce a náhodné útoky.

```
public void work(String workType)
```

Metoda vypočítá dobu časového pásma práce, která je předána argumentem workType a vytváří případný souboj. Dobu vypočítá podle hodnoty zátěže hráče, rychlosti vznášedla, zvoleného herního módu a případně použitého nástroje. Souboj vytváří podle toho, o jaký typ práce se jedná. Během souboje hrají roli vlastnosti vybavení hráče.

Shop.java

Třída určená pro vytvoření objektu, který představuje herní obchod.

TimeChecker.java

Třída vytvořená pro vlákno.

```
Override public void run()
```

Metoda po uplynutí času, kterou udává konstanta THREAD_TIME třídy Constants, volá metodu třídy GameModel, aby ubral hodnoty životních potřeb hráče. Zároveň kontroluje, jestli na planetu Tarrys už nemá přiletět křižník. Metoda běží v cyklu do doby, dokud hráč nepřejde do hlavní nabídky.

Tools.java

Třída obsahující metody, které využívá více tříd.

Závěr

V této práci byla vytvořena desktopová verze karetní hry ve stylu Pazaak s prvky RPG a tematikou Hvězdných válek. Podporuje vývoj možností hráče v čase, zejména co se týče postupného obohacování herního balíčku a vybavení hráče. Hra implementuje algoritmus počítačového protihráče, který je schopný analyzovat situaci a vytvořit strategii během karetní partie. Důležitým prvkem je líbivá vizuální stránka výsledné hry. Hráči jistě ocení uživatelské rozhraní a poutavou hratelnost, kterou hra Paziaatus nabízí.

V rámci bakalářské práce jsem se seznámil a naučil pracovat s modelem DALL-E 2 a grafickým editorem Photopea, které značně obohatili mé grafické dovednosti. Zdokonalil jsem se v objektově orientovaném programování a hlavně ve vývoji desktopových her.

V další fázi vývoje hry plánuji rozšířit podporu na ostatní operační systémy a dodělat funkcionalitu artefaktů a implantátů.

Conclusions

In this work, a desktop version of a Pazaak-style card game with RPG elements and a Star Wars theme was created. It supports the development of the player's options over time, especially regarding the gradual enrichment of the game package and the player's equipment. The game implements a computer opponent algorithm that is able to analyze the situation and create a strategy during the card game. An important element is the pleasing visual aspect of the resulting game. Players will surely appreciate the user interface and engaging gameplay that Pazaak game offers.

As part of my bachelor's thesis, I got to know and learned to work with the DALL-E 2 model and the graphic editor Photopea, which greatly enriched my graphic skills. I have improved in object-oriented programming and mainly in the development of desktop games.

In the next phase of game development, I plan to extend support to other operating systems and complete the functionality of artifacts and implants.

A Obsah příloženého adresáře

game/

Adresář obsahující hru Paziaatus. Hra je spustitelná bez jakékoliv úpravy.

src/

Adresář s kompletními zdrojovými texty hry Paziaatus.

doc.pdf

Text bakalářské práce.

readme.txt

Instrukce pro spuštění hry Paziaatus, včetně všech požadavků pro jeho bezproblémový provoz.

Reference

- [1] Gladius.[online]. citace 20.04.2023.
<https://gladius.gamerz-bg.com/>
- [2] SWTOR - Game Guide. [online]. citace 01.05.2023.
<https://game-guide.fr/98003-swtor-une-petite-partie-de-pazaak/>
- [3] DeviantArt.[online]. citace 01.05.2023
<https://www.deviantart.com/uri101love/gallery>
- [4] DALL · E 2.[online]. citace 01.11.2022
<https://openai.com/product/dall-e-2/>
- [5] Photopea.[online]. citace 15.12.2022
<https://www.photopea.com/learn/>
- [6] Textury RED. [online]. citace 01.04.2023
<https://www.brusheezy.com/textures/58771-5-red-textures>
- [7] Rodina písma Sofachrome. [online]. citace 13.01.2023
<https://www.1001fonts.com/sofachrome-font.html>
- [8] Ikony. [online]. citace 03.02.2023
<https://sites.google.com/site/gson/gson-user-guide>
- [9] Introduction to FXML | JavaFX. [online]. citace 01.05.2023.
https://openjfx.io/javadoc/20/javafx.fxml/javafx/fxml/doc-files/introduction_to_fxml.html
- [10] Java Platform Standard Edition 8 Documentation. [online]. citace 15.02.2023
<https://docs.oracle.com/javase/8/docs/>
- [11] Gson User Guide. [online]. citace 02.02.2023
<https://sites.google.com/site/gson/gson-user-guide>
- [12] Martin, R. C. (2008). Clean Code: A Handbook of Agile Software Craftsmanship. Pearson.