



Gymnázium, Praha 6, Arabská 14

Obor programování, vyučující Mgr. Jan Lana



2D Roguelike hra

Petr Dobiáš

Duben 2024

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská¹⁴ oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V..... dne

Petr Dobiáš

Obsah

1. ANOTACE	4
1. ANOTATION	4
2. CO JE TO ROGUELIKE	5
2.1. PROCEDUÁLNÍ GENERACE LEVELŮ	5
2.2. PERMANENTNÍ SMRT	5
3. POUŽITÉ TECHNOLOGIE	6
3.1. GODOT ENGINE.....	6
3.1.1. <i>Na co jsem Godot použil</i>	<i>6</i>
3.2. ASEPRITE	7
3.2.1. <i>Problém s tvorbou animací</i>	<i>8</i>
4. OVLÁDÁNÍ HRY	9
4.1. POHYB.....	9
4.2. ÚTOK	9
5. PRŮBĚH HRY	10
5.1. STARTOVNÍ MÍSTNOST	10
5.2. STANDARDNÍ HERNÍ MÍSTNOST	11
5.3. MÍSTNOST S BOSSEM.....	12
6. ZÁVĚR	13
7. SEZNAM ZDROJŮ	14
8. SEZNAM OBRÁZKŮ.....	15

1. Anotace

Hráč se pohybuje vygenerovaným světem, který je rozdělený na úrovně. Úkolem hráče je přežít co nejdéle a případně zabít finálního bossa. V průběhu každé úrovně čelí hráč nepřátelům, které musí zabít a tak se přes ně dostat. S nepřáteli hráč bojuje pomocí zbraní, které může sbírat během hraní daného kola hry.

Pokud hráč zemře, musí hru začít od začátku. Díky tomu hráč dostává motivaci přežít co nejdéle a dosáhnout co nejlepšího výsledku.

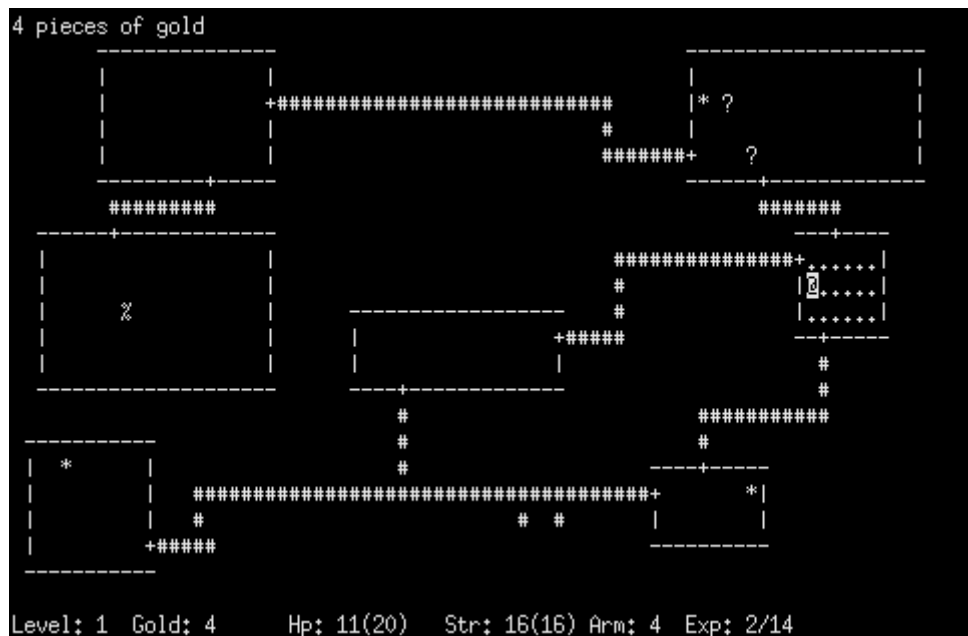
1. Anotation

The player moves through a generated world, which is divided into levels. The player's task is to survive for as long as possible and possibly kill the final boss. During each level you face enemies that you have to kill and overcome. Enemies are fought by the player using weapons that they can collect while playing a given round of the game.

If a player dies, they must start the game from the beginning. Due to this, the player is motivated to survive for as long as possible and achieve the best possible result.

2. Co je to roguelike

Roguelike je jeden z nejstarších herních žánrů. První hra tohoto žánru byla v roce 1980 hra Rogue, podle které je žánr pojmenovaný. hlavními herními mechanikami žánru roguelike jsou procedurálně generované levely a permanentní smrt. V tuto chvíli je hra Rogue k dispozici k zakoupení na platformě Steam za 2,99€ (v přepočtu na české koruny s kurzem ke dni 18. 4. 2024 za 76 Kč)



Obrázek 1 Hra Rogue

2.1. Proceduální generace levelů

Tento prvek je nejdůležitější pro pocit ze hry. Jelikož se hra stále opakuje, musí být nějak zajištěna autentičnost každého kola. To je právě zařízeno proceduální generací levelů. Hra tak není tolik monotónní a každé kolo je jiné.

Před začátkem každého kola se vygeneruje buď z předem definovaných místností, nebo se vygenerují i samotné místnosti.

2.2. Permanentní smrt

Permanentní smrt znamená, že pokud hráč zemře, není žádný způsob jak postavu oživit. Neexistují žádné úložné pozice. To má za příčinu opatrné chování hráče, jelikož spolu se smrtí postavy přijde jeho úsilí na zmar a hráč přijde o postup ve hře.

3. Použité technologie

3.1. Godot Engine



Obrázek 2 Godot Engine logo

Godot Engine je open source, bezplatný herní engine, umožňující tvorbu 2D a 3D her pro různé platformy. Umožňuje tvorbu her pro různé operační systémy, jako například Windows, macOS a Linux.

Godot funguje na principu uzlů. Kde každý uzel představuje například hráče, zvuk, objekt nebo pole. Godot je oblíben pro svou uživatelskou přívětivost a možnost volby preferovaného programovacího jazyku. Základním jazykem je GDScript (vlastní programovací jazyk založený na programovacím jazyku python), ale je možné programovat také v C# nebo C++.

Godot je k dispozici ke stažení zdarma na webové stránce godotengine.org.

3.1.1. Na co jsem Godot použil

Godot byl použit jako hlavní vývojové prostředí. Vybral jsem si ho z několika důvodů. Například kvůli možnosti intuitivně si rozdělit objekty hry. Nejprve jsem totiž chtěl hru dělat v programovacím jazyce Python s knihovnou Pygame. Ovšem v již velmi rané fázi vývoje hry jsem zjistil, že pro struktury hry žánru roguelike je přístup knihovny Pygame velmi matoucí a nepřiměřeně složitý. Vývoj hry se velmi rychle stal nepřehledný a přehlucující.

Jako programovací jazyk jsem použil GDScript, jelikož jsem výrazně více seznámený s programovacím jazykem Python než s programovacím jazykem C++ nebo C#. GDScript je totiž velmi podobný programovacímu jazyku Python a jeho syntaxe je identická a pro mě osobně výrazně příjemnější a čitelnější.

Při vývoji hry jsem hodně využíval systému dědičnosti tříd, který mi například umožnil vytvořit jednu šablonu místnosti. A jelikož je základ každé místnosti stejný, lze podle šablony vytvořit libovolné množství místností s libovolným rozmístěním objektů a nepřátel.

Podobný způsob jsem použil i při tvorbě zbraní. V tuto chvíli se ve hře nachází pouze jedna zbraň, jelikož se mi zcela nepodařilo dodělat funkci zbraní. A to z toho důvodu, že se mi nepodařilo zajistit, aby nepřátelé detekovali kolizi se zbraněmi. Šablona zbraní je ale připravena k tomu, aby se daly vytvořit další zbraně. Jsou už vytvořeny i obrázky dalších zbraní. Jedinou zbraní, která se ve hře v tuto chvíli nachází, je lopata. V případě vytvoření nové zbraně stačí pouze nastavit obrázek zbraně.

Systém dědičnosti je použit i v případě nepřátel. V tomto případě stačí přidat animaci nepřítele a nastavit zónu, kterou kolizí s hráčem ubere hráčovy životy.

Jediný větší problém, na který jsem v Godotu narazil, je jeho neintuitivní řešení kolizí. Nedokázal jsem ho plně pochopit, a i po studování dokumentace si nejsem přesně jistý tím, co všechno kolize ovlivňuje. Kolize jsou rozděleny vyhodnocovány na základě kolizních vrstev, díky kterým se dá například zajistit to, že zbraň může zranit jen nepřítele, a ne hráče a nepřítel může zranit jen hráče, a ne jiného nepřítele. Systém vrstev však konzistentně funguje jen v rámci neděděné scény. Pokud se scény dědí, je třeba zahrnout mnohem více faktorů a vyhodnocování kolizí není jednoduché pochopit a je téměř nemožné přijít na to, co zapříčiňuje detekci či ignoraci kolize.

3.2. Aseprite

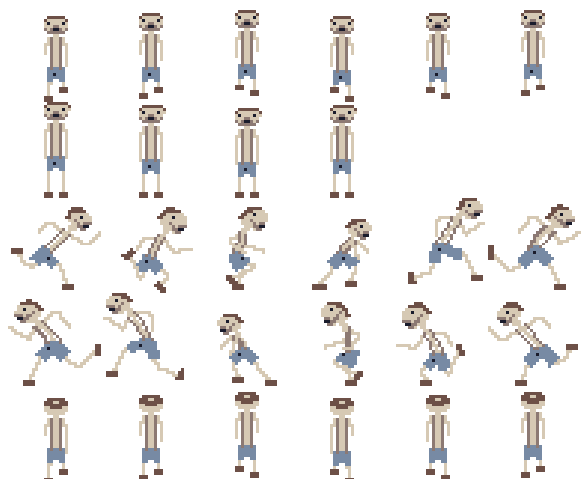


Obrázek 3 Aseprite logo

Aseprite je počítačový program zaměřený hlavně na tvorbu obrázků a animací pro počítačové hry. Je to open source program s velkým počtem vývojářů a rozsáhlou komunitou uživatelů, takže není problém najít návod na jakýkoliv problém. Přímo na hlavní webové stránce programu aseprite.org se nachází velké množství návodů ve formátu videí či textu a obrázků. Tyto návody jsou ovšem často velmi konkrétní a k vyřešení problému často nestačí pouze jeden návod. Dále se na webové stránce nachází odkazy na video návody na platformě youtube.com. Tyto návody jsou třetí strany a z mé zkušenosti jsou v případě Asepritu často užitečnější než návody první strany.

Aseprite je dispozici ke stažení přímo na webu Aseprite za jednorázový poplatek \$19.99 (v přepočtu na české koruny s kurzem ke dni 18. 4. 2024 za 474 Kč), nebo zdarma v repozitáři na GitHub github.com/aseprite. Tento způsob je však poměrně složitý, ale v repozitáři se nachází také návod pro instalaci.

3.2.1. Problém s tvorbou animací



Obrázek 4 Obrázek s animacemi hráče

Aseprite jsem použil k vytvoření všech obrázků a animací hry. Narazil jsem ovšem na jednu nevýhodu. Godot potřebuje pro zobrazení animací soubor, kde jsou snímky animace rozděleny v řadě za sebou viz. Obrázek 4. Aseprite je ovšem schopný zobrazit animace pouze ve vlastním formátu a po exportu do šablony obrázků není schopný převést šablonu zpět. Kvůli tomu nebylo možné jednoduše upravit již exportovanou animaci, a tak se přesvědčit o tom, že snímky animace na sebe navazují. Vždy bylo potřeba manuálně animaci rozdělit na jednotlivé snímky.

4. Ovládání hry

4.1. Pohyb

Hráč se pohybuje stejně, jako ve většině moderních her. A to pomocí kláves **WSAD** následovně:

- Stiskem klávesy **W** nahoru
- Stiskem klávesy **S** dolů
- Stiskem klávesy **A** doleva
- Stiskem klávesy **D** doprava

Další způsob, jakým se může hráč pohybovat, je pomocí herního ovladače. (K dispozici mám pouze Xbox one ovladač, takže nemůžu zaručit, že ovládání bude fungovat i pro ovladače od jiného výrobce.) Hráč se pohybuje pomocí levého joysticku. Tím směrem, kterým joystick nahne, tím se začne pohybovat herní postava.

4.2. Útok

Útok, stejně jako pohyb, je možné ovládat jak pomocí klávesnice, tak pomocí ovladače. Na klávesnici se jedná o šipky. Postava útočí tím směrem, která směrová šipka byla stisknutá. A na (Xbox) ovladači se jedná o tlačítka **XYAB**, které útočí do následovných směrů:

- Stiskem tlačítka **X** doleva
- Stiskem tlačítka **Y** doprava
- Stiskem tlačítka **A** dolů
- Stiskem tlačítka **B** doprava

Toto rozložení ovládání jsem vybral z toho důvodu, že tlačítka **XYAB** směřují do směrů, do kterých postava po jejich stisku útočí. Zprvu bylo útočení na ovladači pomocí pravého horního tlačítka a směr útoku byl určován podle směru naklonění pravého joysticku, ale ovládání pomocí tlačítek **XYAB** je dle mého názoru výrazně jednodušší a intuitivnější.

5. Průběh hry

Po zapnutí hry se ukáže hlavní menu se dvěma tlačítky. Tlačítkem **Quit** zavře herní okno a tlačítkem **Play** se hráč dostane do samotné hry, a to přesně do startovní místnosti. V levém horním rohu se ukazují hráčovy životy. Pokud hráč přijde o všechny své životy, dostane se zpět do hlavního menu.

5.1. Startovní místnost



Obrázek 5 Startovní místnost

Na Obrázku 5 je vidět startovní místnost. Tato startovní místnost je k síti místností připojená svojí levou a spodní stranou. Toto je indikováno zbořeným plotem v levé a dolní části místnosti. Pokud se hráč přiblíží do blízkosti zbořeného plotu, posune se v síti místností tím směrem, na které na které straně se nacházel zbořený plot, ke kterému se přiblížil. Pokud se hráč přiblíží ke straně s nezbořeným plotem, nestane se nic.

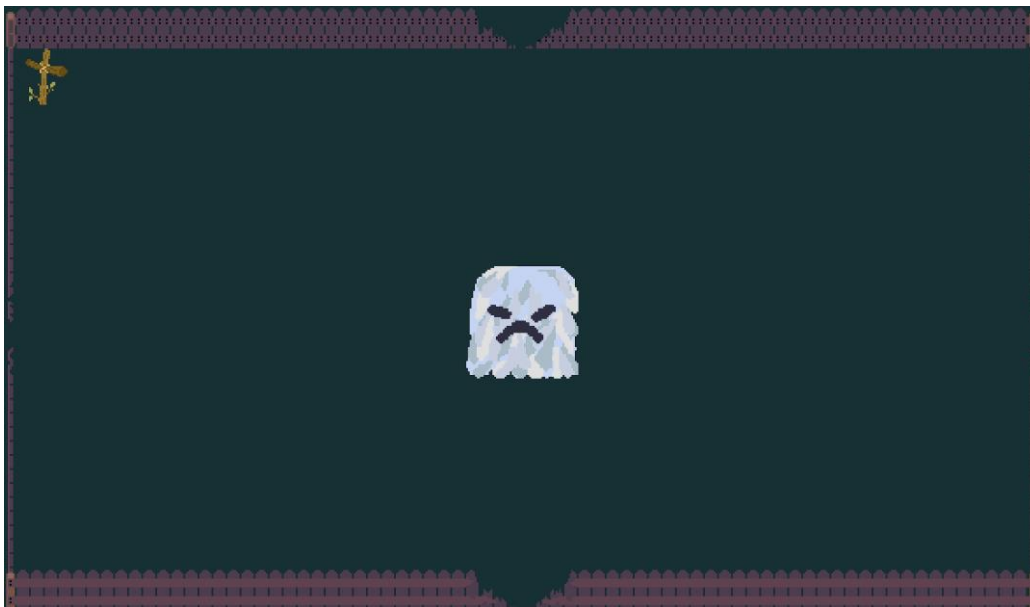
5.2. Standardní herní místnost



Obrázek 6 Místnost se dvěma nepříteli

Na Obrázku 6 je vidět místnost se dvěma nepříteli. Myšlenka hry je taková, že hráč musí zabít všechny nepřátele v místnosti a až poté se může přesunout do další místnosti. Z toho důvodu momentálně místnost vypadá, že není připojena k síti místností žádnou stranu. Pokud hráč ovšem zabije všechny nepřátele, ploty se zboří a hráč se může přesunout do připojených místností. V tuto chvíli ovšem kvůli nekonzistentní detekci kolizí nelze nepřátele zabít, a tak je ve scéně *dungeon* zaškrťovací pole, které po odškrtnutí umožní průchod místnostmi i bez zabití nepřátel. Cesta ke scéně *dungeon* je `res://Scenes/dungeon.tscn`. Zaškrťovací pole se po kliknutí na tlačítko s textem **Dungeon** zobrazí v pravém horním rohu okna jako *Closed If Enemy*. Hráč po kolizi s nepřítelem vždy přijde o jeden život. Jelikož má hráč tři životy, tak po třech kolizích přijde o všechny a dostane se zpět do hlavního menu.

5.3. Místnost s bossem



Obrázek 7 Místnost s bossem

Boss není v tuto chvíli dodělán. Nepohybuje se a nemůže hráče zranit. Zároveň, kvůli nekonzistentnímu vyhodnocování kolizí nemůže být hráčem zabit, tudíž není způsob, jak hru dohrát. Myšlenka pohybu bosse je taková, že se bude pohybovat diagonálně po místnosti. Podobně jako se pohybuje například kulečnicková koule po kulečnickovém stole. Po určitém počtu nárazů do zdi by se boss na chvíli zastavil a objevila by se nad ním animace točících se hvězdiček jako indikace znehybnění bosse kvůli nárazům do zdi. Do té doby by se hráč musel pohybům bosse vyhýbat. Po znehybnění bosse by mohl hráč bosse zranit. Toto by se opakovalo do té doby, dokud by hráč bosse nezabil a tím by hra skončila.

6. Závěr

S projektem jsem poměrně spokojen. I když jsem kvůli nepochopení detekce kolizí nemohl vytvořit podmínku, která by ukončovala hru po výhře. Je pro to ale vše připraveno a pokud se mi podaří opravit detekci kolizí, tak chybí dodělat pouze kód pro pohyb hlavního bosse hry.

Kvůli zdlouhavému řešení problému s kolizemi jsem nestihl přidat více zbraní. Mám už ale promyšlené, které další zbraně by se mohli do hry přidat. K těmto zbraním mám vytvořené už i obrázky. První zbraní by mohl být krumpáč, který by za cenu nižší rychlosti útoku útočil do všech stran tím, že by se kolem hráče při útoku otočil. Další zbraní by mohla být například berle, která by útočila ve stejném stylu jako lopata, ale za cenu síly útoku by útočila rychleji.

Také mám vymyšleného i druhého bosse. Tím by mohl být duší červ, který by střídavě vyskakoval a zase se zavrtával do podlahy místnosti. Někdy by se však nedokázal zavrtat a v tu chvíli by ho hráč mohl zranit.

Myslím si, že vybrat si toto téma nebylo úplně nejlepší rozhodnutí. Vybral jsem s i ho z toho důvodu, že herní žánr je jeden z mých nejoblíbenějších a zajímalo mě, co všechno tvorba hry tohoto žánru obnáší. Během vývoje jsem zjistil, že i když se žánr zdá první pohled z hlediska programování jednoduchý, tak je velmi složitý. Hru jsem také dělal v prostředí, se kterým jsem se dříve vůbec nesetkal, a to mi vývoje ještě o to více ztížilo. Zároveň si ale myslím, že mi dal projekt velké množství zkušeností pro práci s Godotem a přesvědčil jsem se, že i přes to, že je mnohem jednodušší než alternativy jako například Unity, tak je to velmi silný nástroj pro tvorbu her.

7. Seznam zdrojů

1. Roguelike. <https://cs.wikipedia.org/wiki/Roguelike>
2. Godot logo. <https://godotengine.org/assets/logo.svg>
3. Aseprite logo. <https://github.com/aseprite/aseprite/blob/main/data/icons/ase256.png?raw=true>
4. Aseprite dokumentace. <https://github.com/aseprite/aseprite/blob/main/data/icons/ase256.png?raw=true>
5. Godot. <https://godotengine.org/>
6. Godot dokumentace. <https://docs.godotengine.org/en/stable/>
7. Hra Rogue. https://upload.wikimedia.org/wikipedia/commons/thumb/0/0c/Rogue_Screenshot.png/220px-Rogue_Screenshot.png

8. Seznam obrázků

Obrázek 1 Hra Rogue.....	5
Obrázek 2 Godot Engine logo.....	6
Obrázek 3 Aseprite logo	7
Obrázek 4 Obrázek s animacemi hráče	8
Obrázek 5 Startovní místnost	10
Obrázek 6 Místnost se dvěma nepříteli.....	11
Obrázek 7 Místnost s bossem.....	12