

Střední průmyslová škola elektrotechnická

Ječná 30, Praha 2



# Odborná maturitní práce

Praktická zkouška z odborného předmětu

Jan Růžička

Praha 2022

Rogue-like 3D hra s prvky vlastní grafiky

# Project Dungeon

Powered by Unity Engine

Autoři: Jan Růžička, Maxim Kalvoda  
Obor studia: 18-20-M/01 Informační technologie  
Třída: C4b  
Školní rok: 2021/2022  
Vedoucí práce: Mgr. Alena Reichlová

## Prohlášení

Prohlašuji, že v souladu se zadáním jsem na projektu pracoval ve dvojici. Použil jsem pouze zdroje uvedené v této dokumentaci.

Souhlasím, aby naše práce byla volně přístupná k prezenčnímu studiu na SPŠE Ječná.

Dne:

---

Jan Růžička:

---

## Poděkování

Rád bych poděkoval Mgr. Aleně Reichlové za vedení naší dlouhodobé maturitní práce.

# Obsah

Prohlášení.....	3
Poděkování.....	3
Anotace.....	5
Úvod.....	5
Zadání.....	5
Název.....	5
Použitý Software.....	6
Unity.....	6
Blender.....	6
Clip Studio Paint.....	6
Visual Studio Community.....	6
Modely.....	7
Ukázka modelů.....	7
Avatar of the Core.....	7
Horní část.....	7
Spodní část.....	8
Dekorace místnosti.....	8
Animace.....	9
Textury.....	9
Ukázka textur.....	10
Obrázky.....	11
Ukázka obrázků.....	11
Scény.....	12
Levely.....	12
Lobby.....	12
Menu.....	13
Položky.....	13
Loading.....	13
Skripty.....	14
GameData.....	14
GameManager.....	15
GlobaFunctions.....	15
Enemy.....	15
Crystall.....	16
Loot.....	17
Chest.....	17
Options.....	17
Ostatní.....	18
AI.....	19
Avatar of the Core.....	19
Slime.....	20
Wolf.....	20
Zdroje.....	21
Závěr.....	22

# Anotace

Tato dokumentace popisuje Rogue-like 3D hru s prvky vlastní grafiky v herním enginu Unity, včetně zpracování modelů v Blender a skriptů herních mechanik v jazyce C#. Hlavním cílem této práce bylo vytvořit naší první vlastní funkční hru na základě zkušeností ze studia předmětu programování v aplikacích.

## Úvod

### Zadání

Cílem zadání je vytvořit jednoduchou Rogue-like hru za pomoci Unity engine a programovacího jazyka C#, včetně prvků vlastní grafiky. Hráč bude mít za cíl vydržet co nejdéle, při hraní bude moci vylepšovat svoje schopnosti, získávat nové zbraně za splnění úkolů. Jednotlivé prvky hry budou ručně zhotovené v 3D modelovacím softwaru Blender.

### Název

Každé dílo by mělo mít nějaké jméno, které bude reprezentovat hru. Společně s kolegou jsme vymysleli název Project Dungeon, protože sám vypovídá o obsahu naší práce. Slovo Project vyjadřuje soubor aktivit a Dungeon je v překladu jeskyně, což je tajemné místo, které vzbuzuje zvědavost a zájem o prožití dobrodružství.

# Použitý Software

## Unity

Jedná se o multiplatformní herní engine pro vývoj her od firmy Unity Technologies. Byl použit pro vývoj naší hry na PC. Výhodou je možnost jeho využití zcela zdarma, a to pokud výdělek nepřekročí 100 000\$ ročně.

Takže zájemcům o vývoj her doporučuji zvážit použití herního engine Unity. Náš projekt byl vytvořen na verzi 2020.3.15f2. Využili jsme také doplněk v Unity Asset Store: Blockout, což je nástroj pro rychlejší stavbu mapy.

## Blender

Jedná se o bezplatný open-source, a to i pro komerční využití, pod licencí GNU GPL. Je to modelovací program pro vytváření 3D grafiky a animaci s využitím různých technik, např. světla, atd. Blender je renderován pomocí knihovny OpenGL, díky které lze snadno přenést všechny modely vytvořené v Blenderu na všechny podporované platformy, např. Unity. Byl použit jako hlavní modelovací prostředek pro vytvoření základních modelů do hry.

## Clip Studio Paint

Jedná se o placený program od společnosti Celcys. Je to profesionální grafický editor pro kreslení 2D grafiky, obecné ilustrace a tvorbu komiksů. Podporuje 2D animace, bitmapové a vektorové kreslení textu. Byl použit hlavně pro vytvoření ikoněk předmětů, UI inventáře a dalších doplňků do hry.

## Visual Studio Community

Jedná se o IDE od společnosti Microsoft, což je vývojové prostředí pro programování. Toto IDE je zcela zdarma a podporuje mnoho programovacích jazyků, např. C#, C++, atd. Má v sobě zabudovanou funkci IntelliSense, která pomáhá opravit chyby v kódu a také chytře nabízí nejlepší možnosti při dokončení kódu.

# Modely

S kolegou jsme se rozhodli, že grafika bude mít jednoduché tvary. Tento styl nemá v sobě velké detaily, díky tomu nemá často velký počet bodů polygonu, ale zároveň stačí k tomu, aby na první pohled bylo zřejmé o jaký model se jedná.

Při výběru tohoto stylu nás inspirovala hra Genshin Impact, která vyšla v roce 2020, hrál jsem ji a velice se mi líbila. Navíc je tento styl velmi jednoduchý na tvorbu.

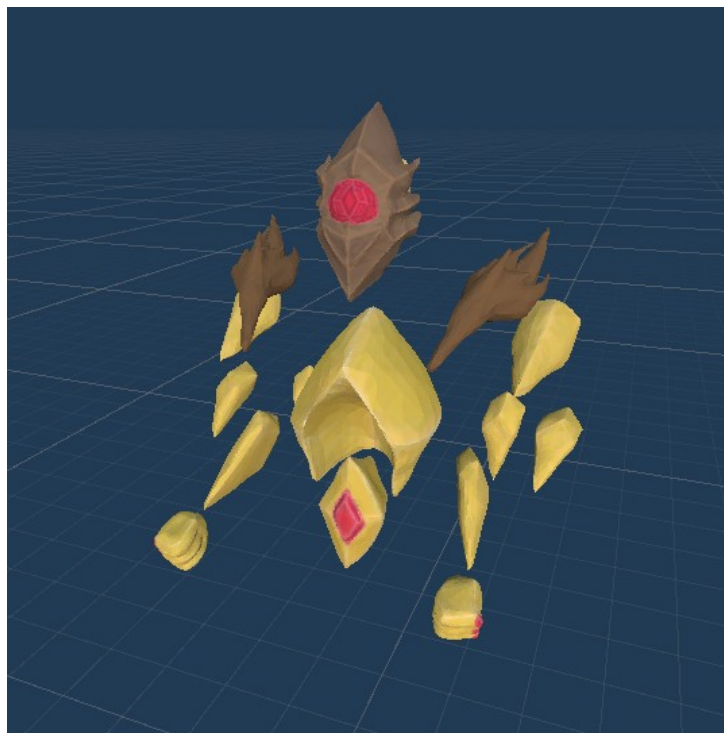
Díky malému počtu bodů polygonu se hra stává plynulejší a zároveň můžeme využít více modelů najednou.

## Ukázka modelů

### Avatar of the Core

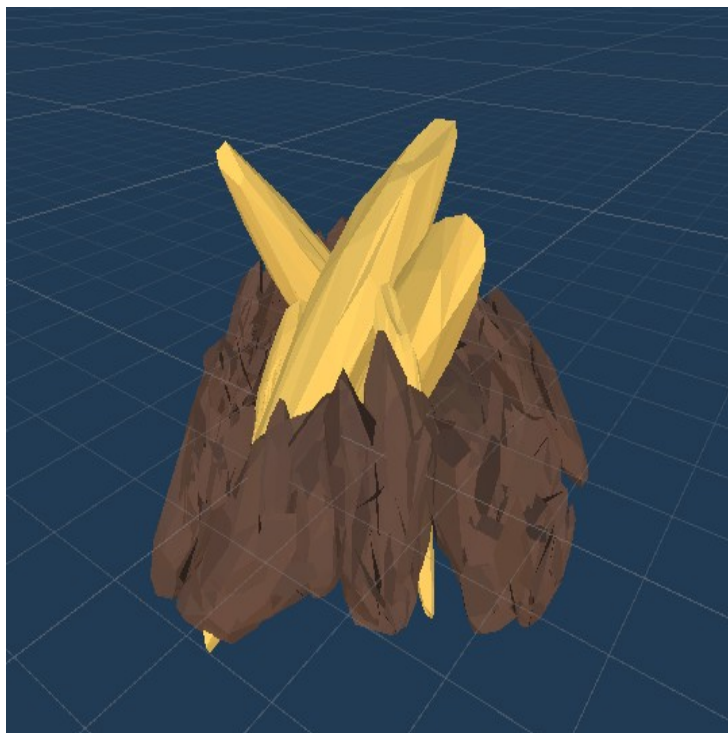
Rozhodl jsem se rozdělit model na 2 části, horní a spodní. Je to kvůli tomu, že během boje je nehybný a otáčí pouze hlavním tělem.

#### *Horní část*



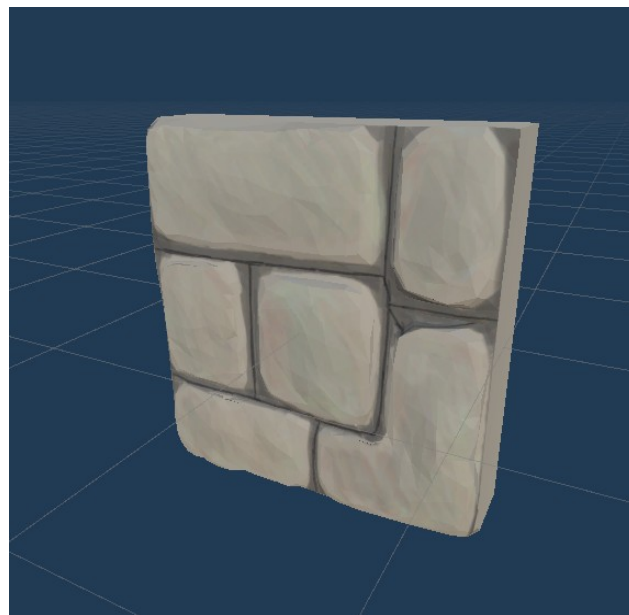
Základní model hlavního těla Avatar of the Core

## *Spodní část*



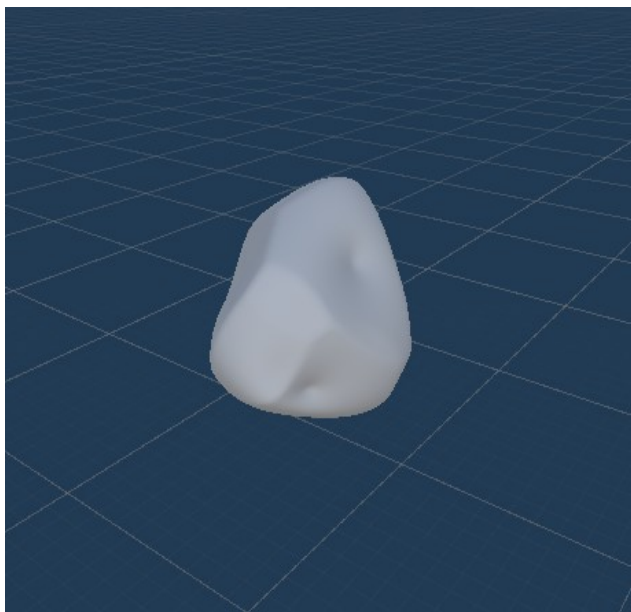
Základní model spodního těla Avatar of the Core

## **Dekorace místnosti**

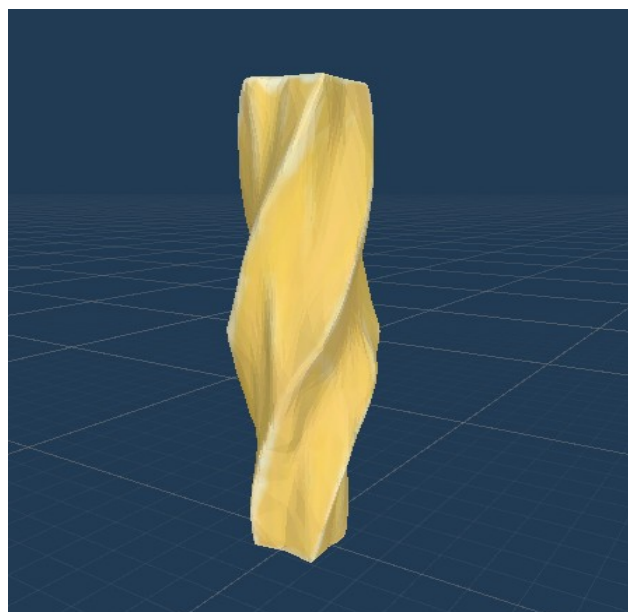


Základní modely stěn





Základní model kamene



Základní model sloupu

## Animace

Z časových důvodů ve hře nebyly použity žádné animace.

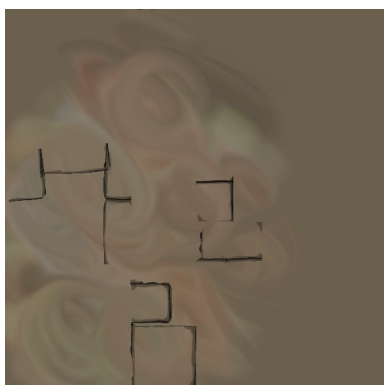
## Textury

Ihned po vytvoření modelu, přejdeme ze Sculpt/Object modu do Edit modu. Tam si označíme všechny důležité okraje pomocí funkce Mark Seam. Jako důležité okraje beru pro příklad hrany z krychle nebo hrany kolem důležitých částí, jako jsou třeba oči. Po označení všech důležitých hran stiskneme klávesu A (pro označení všech bodů/hran na našem modelu) a ihned potom klávesu U (pro otevření menu možností pro Unwrap funkci) a zvolíme základní funkci Unwrap. Tato funkce funguje následovně. Pomocí hran, které jsme označili dokáže “promítnout” svůj povrch do 2D obrázku, na kterých se dané zvýrazněné hrany ukáží. Pak přejdeme do z Edit modu do Paint modu, tam si jako první věc vytvoříme materiál pro náš model. Po vytvoření materiálu už můžeme začít s tvorbou textur, když jsme hotovi, tak texturu uložíme jako samostatný obrázek.

## Ukázka textur



Ukázky textur pro model Avatar of the Core



Ukázky textur pro model stěn

# Obrázky

Obrázky (v angličtině Sprite) jsem hlavně používal pro ikony předmětů. Vytvářel jsem je pomocí softwaru Clip Studio Paint.

## Ukázka obrázků



Ukázka obrázku meče



Ukázka obrázku dýky



Ukázka obrázku života hráče



Ukázka obrázku many hráče

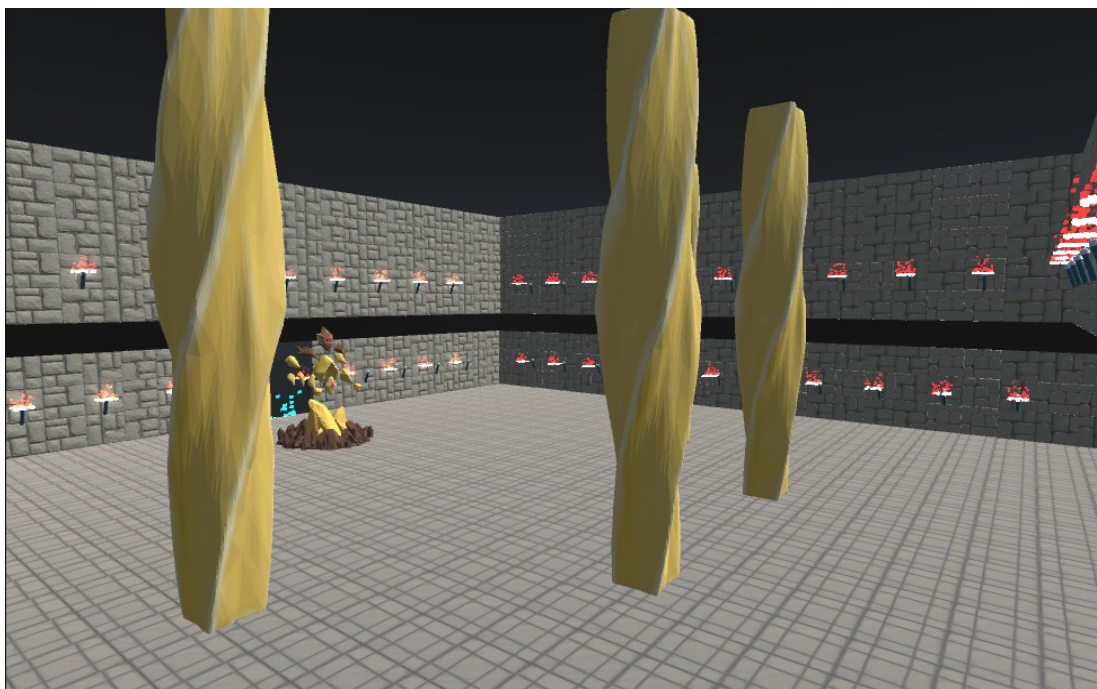
## Scény

Hra obsahuje 7 scén, které jsem rozdělil na skupiny – 4 levelů, 1 lobby, 1 menu a 1 loading. Jednotlivé scény se otevírají pomocí skriptů.

## Levely

Level je předem postavený pomocí editoru Unity. Na začátku hry se náhodně vytvoří pořadí jednotlivých levelů a hráč je postupně musí projít. Na konci každého levelu se nachází portál, kterým se přesune do dalšího levelu. Na posledním levelu čeká silný protivník Avatar of the core, se kterým hráč svede poslední bitvu. Po jeho porážce se otevře vítězné okno: „Congratulations you win.“

Pokud hráč v průběhu hry nedokáže projít některý level, aktivuje se animace „Death“ a hráč se vrátí zpátky do scény lobby.



Ukázka finálního levelu

## Lobby

Ve scéně lobby se nachází:

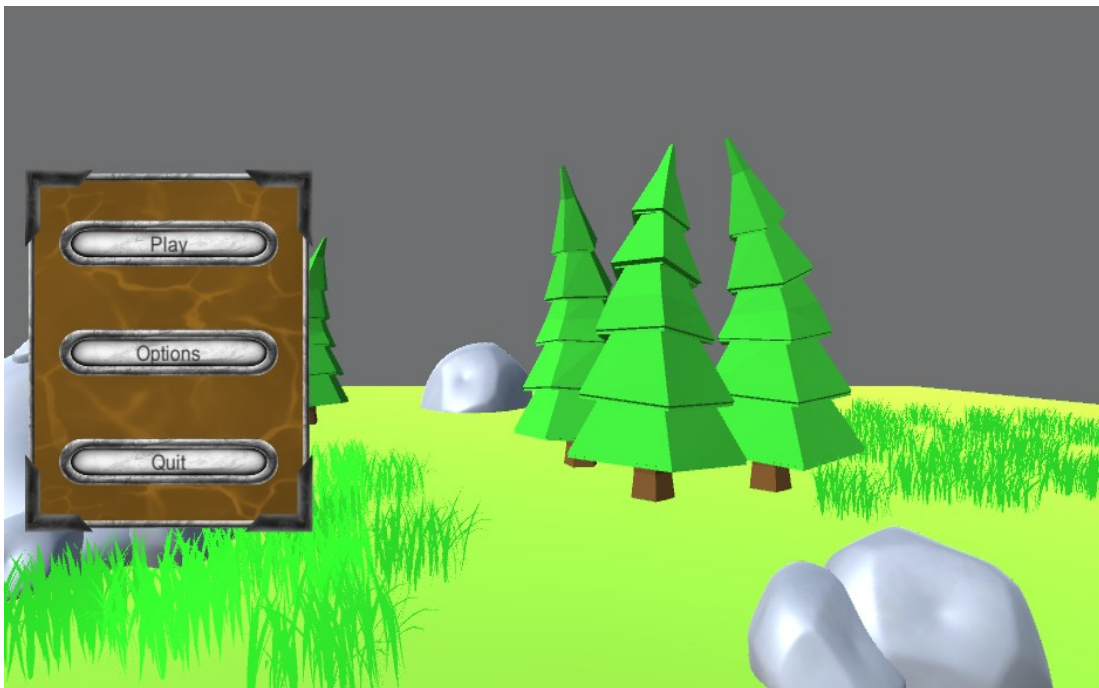
- Start prvního levelu
- 2 obchody, ve kterých může hráč nakupovat zbraně a lektvary
- Cech, kde může hráč vybírat úkoly
- Místo k vylepšování schopností nebo výběr zbraní.

## Menu

Při spuštění hry se automaticky otevře scéna menu (rozcestník). Hráč může zapnout hru, jít do nastavení nebo vypnout hru.

### Položky

- Play – zapnutí
- Options – nastavení klávesnice, rozlišení a grafiky
- Quit – ukončení



Ukázka menu

## Loading

Při zapnutí scény se automaticky vybere další scéna, která se později spustí.

# Skripty

Hrají největší roli v Unity. Skripty obsahují veškerou herní logiku včetně mechanik a průběhu hry.

Skripty jsou psané v jazyce C#, musí být v Unity vždy připojena k objektu. Existují výjimky, kdy skripty nemusí být připojena a slouží jako aktivace funkce, např. ke spočítání nějaké hodnoty. Skriptování je řada instrukcí pro danou aplikaci, na rozdíl od programování, při kterém programátor poskytuje sadu pokynů pro počítač k dosažení určitého cíle.

V Unity jsem vytvořil přibližně 30 skriptů, některá se níže pokusím popsat.

## GameData

Jedná se o jeden nejdůležitějších skriptů, pomocí něhož ukládáme a zapisujeme všechny informace o hráči. Jedná se o statickou třídu, která v sobě drží všechna důležitá data o aktuálním stavu hry. Tento aktuální stav dokáže načítat a ukládat do souboru ve formátu JSON.

## Proměnné

- Peníze hráče.
- Zkušenosti hráče.
- Level hráče.
- Level lektvarů hráče.
- Počet dovednostních bodů hráče.
- Správa úkolů hráče.
- Správa zbraní, které hráč vlastní.
- Správa schopností hráče.
- Správa předmětů z inventáře hráče.

## Metody

- Unlock - slouží k odemykání schopností a následně zapíše do JSON souboru.
- UnlockWeapon - slouží k odemykání zbraní a následně zapíše do JSON souboru.
- CheckMission - slouží ke zjištění, zda hráč splnil úkol.
- ResetMission - smaže všechny aktivní úkoly.
- AddXp - přidává zkušenosti hráči.
- LoadGameData - načte z JSON souboru data, která byla uložena.
- SaveGameData - uloží data JSON souboru.

## GameManager

Slouží k ovládání pause menu a na začátku levelu vždy připraví scénu (načte GameData, připraví inventář, nastaví schopnosti)

### Proměnné

- Odkaz na hlavní UI.
- Odkaz na nastavovací UI.
- Odkaz na zbraň, kterou hráč právě drží.

### Metody

- Resume – vrátí hráče zpět do hry.
- Options – otevře nastavení.
- Back – vrátí hráče zpět do pause menu.
- Quit – ukončí hru.

## GlobaFunctions

Jedná se o moji speciální knihovnu, ve které se nacházejí všechny funkce pro vyvolání určitých výpočtů. Aby se daly funkce využít, je potřeba mít předem nastavené hodnoty – viz. obrázek níže. Vlastní knihovnu jsem si vytvořil, abych nemusel při každém opakovaném použití znovu psát stejný kód. Takže si ho jen vyvolám, což je rychlejší.

```
Počet odkazů: 7
public static float WaitTime(float minTime, float maxTime)
{
    float waitTime = Random.Range(minTime, maxTime);

    return waitTime;
}
Počet odkazů: 2
```

Ukázka funkce z několika funkcí

## Enemy

Tento skript spravuje funkce a proměnné protivníka. Jedná se o abstraktní třídu, která je následníkem třídy MonoBehaviour od Unity. Slouží hlavně jako základ dědičných vlastností pro následníky třídy pod skupinou enemy.

## Proměnné

- Životy protivníka.
- Síla útoku protivníka.
- Čas, kdy může protivník znovu útočit – dělíme na min. a max.
- Zkušenosti, které zbydou od poraženého protivníka.
- Peníze, které zbydou od poraženého protivníka.
- Jméno protivníka.

## Metody

- TakeDamage – ubírá protivníkovi životy, pokud mu propadnou všechny, umře a tím přibudou hráčovi peníze a zkušenosti.
- ApplyForce – odhazuje protivníka po úderu od hráče.
- LookAtPlayer – protivník sleduje hráče.

## Crystall

Jedná se zkrácený název Avatar of the Core. Boss nebo-li protivník dědí vlastnosti od třídy Enemy a k tomu má ještě další proměnné a metody, které spravují funkci protivníka

## Proměnné

- Všechny základní proměnné z třídy Enemy.
- Síla útoku laseru protivníka.
- Etapy typu Enums – celkem 3 etapy.
- Seznam protivníků, které protivník může vyvolat.
- Seznam všech bodů pro vyvolání protivníka.

## Metody

- Všechny základní metody z třídy Enemy.
- CheckStage – hlídá, zda protivník nemá jít do další etapy.
- NextStage – postup do další etapy.
- Die – smrt protivníka.
- SpawnShield – přidá protivníkovi štít, aktivuje se po další etapě.
- SpawnEnemies – vybere protivníka ze seznamu a vyvolá ho.
- LaserAttack – připraví laserový útok a poté vystřelí po hráči.
- RangeAttack – střílí na dálku.

## Loot

Tento skript pracuje na stejném principu jako skript Enemy.



## Proměnné

- Zkušenosti z pokladu.
- Peníze z pokladu.

## Metody

- AddXp – přidá hráči zkušenosti.
- AddMoney – přidá hráči peníze.

## Chest

Tento skript pracuje na stejném principu jako skript Crystall.

## Proměnné

- Všechny základní proměnné z třídy Loot.
- Proměnná typu bool, zda je hráč v blízkosti truhly.
- Proměnná typu bool, zda byla truhla otevřena.

## Metody

- OnTriggerEnter – kontroluje, zda je hráč poblíž truhly.
- OnTriggerExit – kontroluje, zda není hráč poblíž truhly.

## Options

Tento script je pro ovládání nastavení v main menu a pause menu. Dokáže nastavit grafiku, zvuky a ovládání hry. Celé nastavení dokáže načítat a ukládat do souboru ve formátu JSON.

## Metody

- SetResolution - nastaví rozlišení.
- SetQuality - nastaví kvalitu hry.
- KeySettings - otevře key-settings panel.
- SetKey - nastaví tlačítko k určité akci (skok, útok, atd.).
- Back - zavře key-settings panel.
- Save - uloží aktuální nastavení do souboru.
- Load - načte aktuální nastavení ze souboru.
- SetMasterVolume, SetSfxVolume, SetMusicVolume - nastaví hlasitost zvuku.

## Ostatní

Myslím si, že podrobně popisovat všechny skripty je zbytečné, takže uvádím níže souhrn všech skriptů a jejich stručný popis.

- Arrow – skript, který spravuje pasti pro střelbu šípů.
- Destroying – skript, který ovládá trajektorii střely.
- Enter – skript, který sleduje, jestli je hráč v blízkosti protivníka.
- HealthBar – skript, který ukazuje stav životů protivníka.
- IBreakable – rozhraní, které implementuje metodu pro okamžité zničení objektu.
- ILootBreakable – rozhraní, které implementuje metodu pro postupné zničení objektu.
- MainMenu – skript, který slouží pro ovládání hlavního menu na začátku hry.
- MyData – tato třída slouží pro uložení všech informací, které se následně zapíší do JSON souboru.
- SceneLoading – skript, který slouží pro načtení další scény.
- SceneGame – skript, který vybere další level.
- Shield – skript, který umožňuje použít štít protivníkovi.
- ShotArrow – skript, který ovládá trajektorii šípu.
- Slime – skript spravující funkce protivníka Slime.
- Spawn – skript, který po čase přivolá protivníky do hry.
- SpawnEnemies – skript, který na začátku hry přivolá určitý počet protivníků.
- SpawnLoot – skript, který na začátku hry vloží poklady s určitou šancí.
- Spike – skript spravující funkce pasti Spike.
- Start\_Intro\_Crystall – stavový skript pro Avatar of the Core, který navazuje na skript Enter.
- Start\_Follow\_Crystall – stavový skript pro Avatar of the Core, který vybírá útoky a sleduje hráče.
- Start\_Attack\_Slime – stavový skript pro Slime, který sleduje, zda může Slime zaútočit.
- Start\_Follow\_Slime – stavový skript pro Slime, který navazuje na skript Enter.
- Start\_Follow\_Wolf – stavový skript pro Wolf, který navazuje na skript Enter.
- Start\_Idle\_Wolf – stavový skript pro Wolf, který přepíná do Idle a Patrolling stavu.
- Start\_Intro\_Wolf – stavový skript pro Wolf, který navazuje na skript Enter.
- Trap – skript spravující základní funkce a proměnné pro pasti.
- Vase – skript spravující funkce pokladu Vase.
- Wolf – skript spravující funkce protivníka Wolf.
- WeaponData – tato třída slouží pro uložení všech informací o zbraní, kterou zrovna drží.

# AI

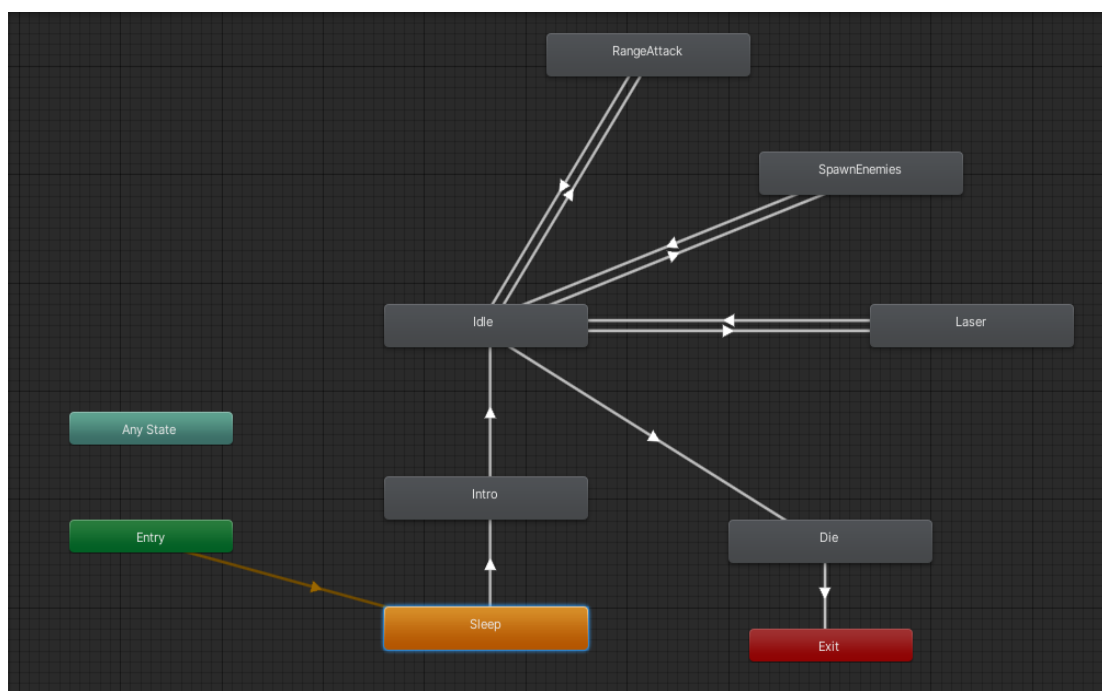
Umělou inteligenci jsem dělal pomocí State Machine (v češtině stavový stroj). Jedná se o koncept, který uchovává stav v daném okamžiku. Díky tomu bylo jednodušší vytvářet umělou inteligenci.

## Avatar of the Core

Protivník se nachází ve stavu Sleep do doby, než ho naruší hráč v jeho prostoru. Jakmile se probudí, přepne se do stavu Intro, dohraje animaci a následně dostává do stavu Idle. Ve stavu Idle vybírá druh útoku, přitom sleduje hráče a po určitém čase zaútočí.

Avatar of the Core zpočátku může používat jenom jeden druh útoku. Aby mohl protivník využívat další druhy útoků, musí mu klesnout počet životů pod určitou hranici a potom vstoupí do další etapy.

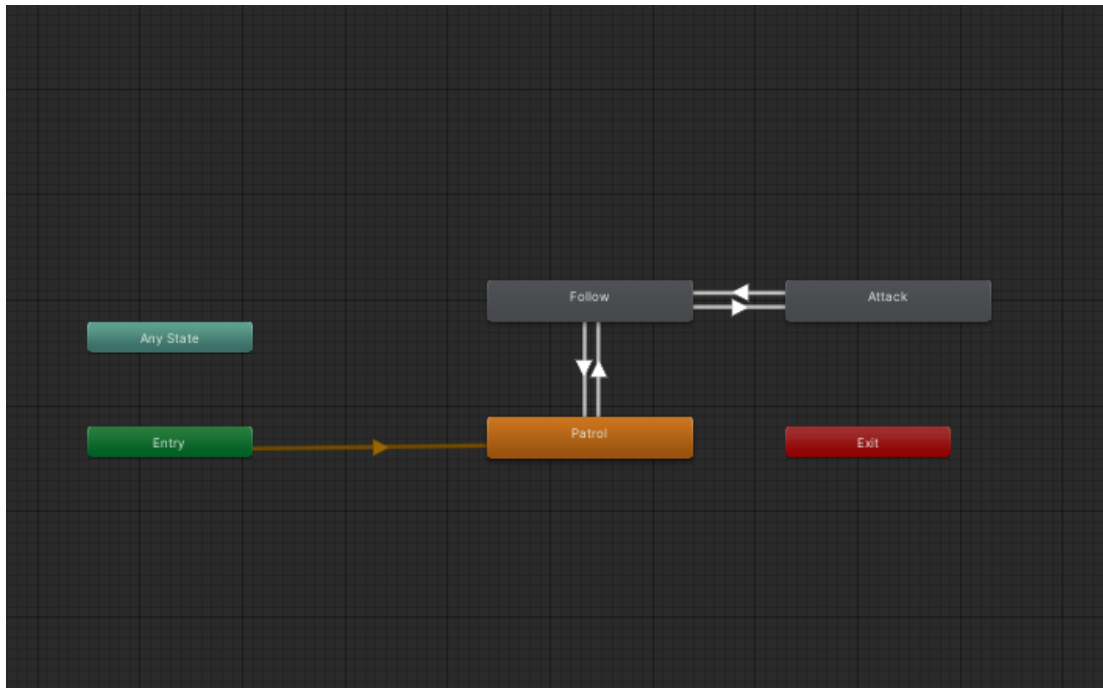
Jakmile vstoupí do další etapy, odemkne se mu druh útoku SpawnEnemies, kterým přivolá dalšího slabšího protivníka. Po další etapě může používat druh útoku Laser. Ovšem nabití laseru trvá dlouho, proto se musí hráč schovat pod nějaký objekt, aby se ubránil útoku. Jedná se o nejsilnější útok protivníka.



Ukázka State Machine pro Avatar of the Core

## Slime

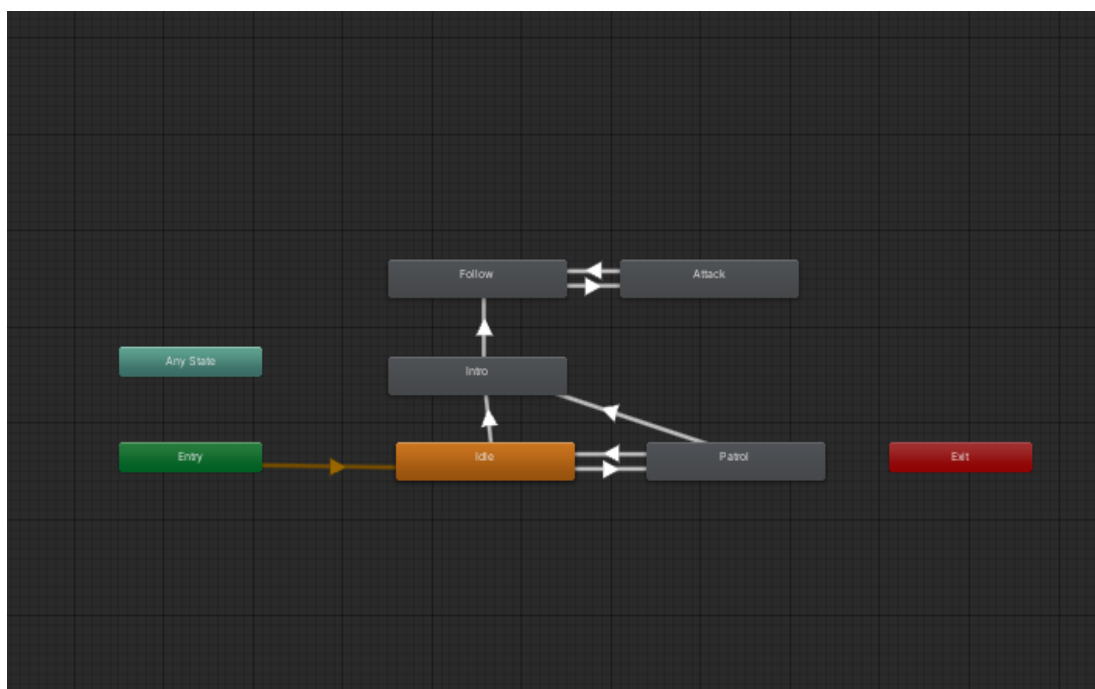
Protivník prochází po mapě tak dlouho, dokud nenajde hráče. Jakmile hráče potká, začne ho pronásledovat. Jakmile se přiblíží k hráči, aktivuje se stav Attack. Pokud se mu útok povede, sníží hráčovi životy. Pokud se hráčovi podaří uniknout, přestane ho protivník pronásledovat a může se znovu procházet po mapě.



Ukázka State Machine pro Slime

## Wolf

Tento protivník je výjimečný oproti Slime, má 2 stavy – Patrol a Idle. Při stavu Patrol prochází po mapě. Při stavu Idle zůstává protivník na stejném místě. V případě, že potká hráče, bude ho pronásledovat, i přesto že ho ztratil. Takže je potřeba být velice opatrný.



Ukázka State Machine pro Wolf

## Zdroje

Během práce jsem čerpal z několika zdrojů.

### Použitý font

- Linux Biolinum G

### Fóra při řešení případných problémů

- <https://stackoverflow.com/>
- <https://forum.unity.com/>

### Kanály, kterými jsem se inspiroval

- <https://www.youtube.com/c/Brackeys>
- <https://www.youtube.com/c/CodeMonkeyUnity>
- <https://www.youtube.com/c/unity>

## Dokumentace k využitým softwarům

- <https://docs.unity3d.com/Manual/index.html>
- <https://docs.microsoft.com/cs-cz/>
- <https://www.clipstudio.net/en/manuals/>
- <https://docs.blender.org/manual/en/latest/index.html>

## Třetí knihovna při práci v Unity

- <https://assetstore.unity.com/packages/tools/level-design/blockout-100388>

## Závěr

Touto prací jsem se pokusil popsat fungování hry v herním enginu Unity. Shrnujím všechny důležité části a funkce hry.

Mým cílem je vytvořit hru, kterou bych v budoucnu mohl prodávat. Tato problematika mě zajímá a to je také důvod, proč jsem nastoupil na školu s oborem informatika. Díky dlouhodobé maturitní práci jsem se naučil základy vývoje počítačové hry. Tato práce mi také pomohla zlepšit své dovednosti při modelování v Blenderu a kreslit textury a obrázky.

Hru budu nadále s kolegou vylepšovat, dokud nebude tak na takové úrovni, aby byla vhodná pro hráče jako jsem já. Mrzí mě, že jsem nestihl vymodelovat více objektů a neudělal žádné animace.