

Odborná maturitní práce



Střední průmyslová škola elektrotechnická
Ječná 30, Praha 2

ODBORNÁ MATURITNÍ PRÁCE

Rogue-like 3D hra s prvky vlastní grafiky

Project Dungeon

Autor: Maxim Kalvoda

Škola: SPŠE Ječná, Ječná 30, 121 36, Praha 2

Obor studia: 18-20-M/01 Informační technologie

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:

Maxim Kalvoda:

Poděkování

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

Obsah:

- **Anotace**
- **Úvod**
 - Zadání
 - Název
- **Použitý Software**
 - Unity
 - Visual Studio Community
- **Scény**
 - Main Menu
 - Lobby
- **Animace**
 - Zbraně
 - UI
 - Ostatní
- **User Interface**
 - Nastavení
 - Menu
 - Inventář
 - Obchody
- **Audio**
 - Audio Mixer
 - Hudba a zvukové efekty
- **Použité knihovny**
 - JsonDotNet
- **Scripty**
 - Audio
 - Inventář
 - Mise
 - Hráč
 - Obchody
 - Ostatní
- **zdroje**
- **závěr**

Anotace:

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

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 splněné úkoly. 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 kolegy 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

Unity je herní engine od společnosti Unity Technologies, ve kterém byl náš projekt vytvořen. Náš projekt byl vytvořen primárně pro PC, ale umožňuje také tvorbu her pro konzole, mobily, web a virtuální realitu. Unity poskytuje možnosti vývoje jak 2D tak i 3D her. Kromě grafického prostředí pro tvorbu, podporuje také tvorbu skriptů v jazyce C#. Náš projekt byl vytvořen ve verzi 2020.3.15f2

Visual Studio Community

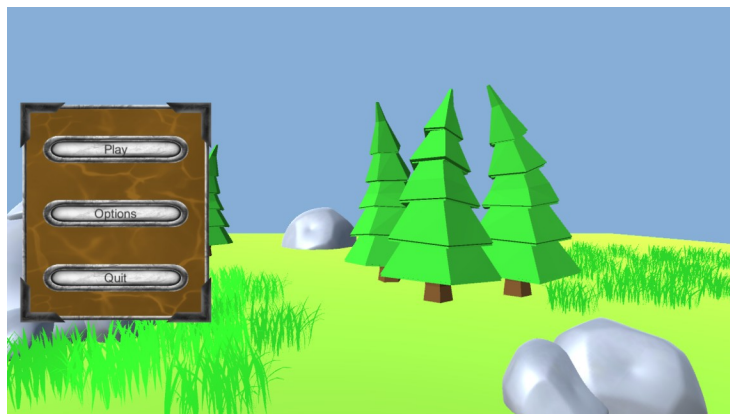
Microsoft Visual Studio je vývojové prostředí od Microsoftu, které jsme použili pro skriptování v jazyce C#. Visual Studio obsahuje editor kódu podporující IntelliSense (nabízí možnosti pro dokončení kódu) a je plně propojeno s Unity a tak usnadňuje vývoj projektu.

Scény:

Main menu

Je to první scéna, která se po zapnutí hry objeví. Obsahuje 3 tlačítka a pro lepší atmosféru jsem přidal několik modelů v pozadí. Z menu lze přejít do nastavení, začít hrát a nebo ukončit program. Při přechodu do nastavení se objeví Options menu, kde lze nastavit například hlasitost nebo rozlišení obrazovky.

Main menu:



Lobby

Jde o úvodní lokaci, kam se hráč dostane a kam se bude po každé smrti vracet. Vytvořil jsem ji tak, aby na hráče působila co nejpříznivěji. Nejsou zde žádní nepřátelé, pouze obchody a cesta do levelu. V obchodech se hráč může po každém pokusu vylepšovat, a tak získat výhodu do dalších pokusů.

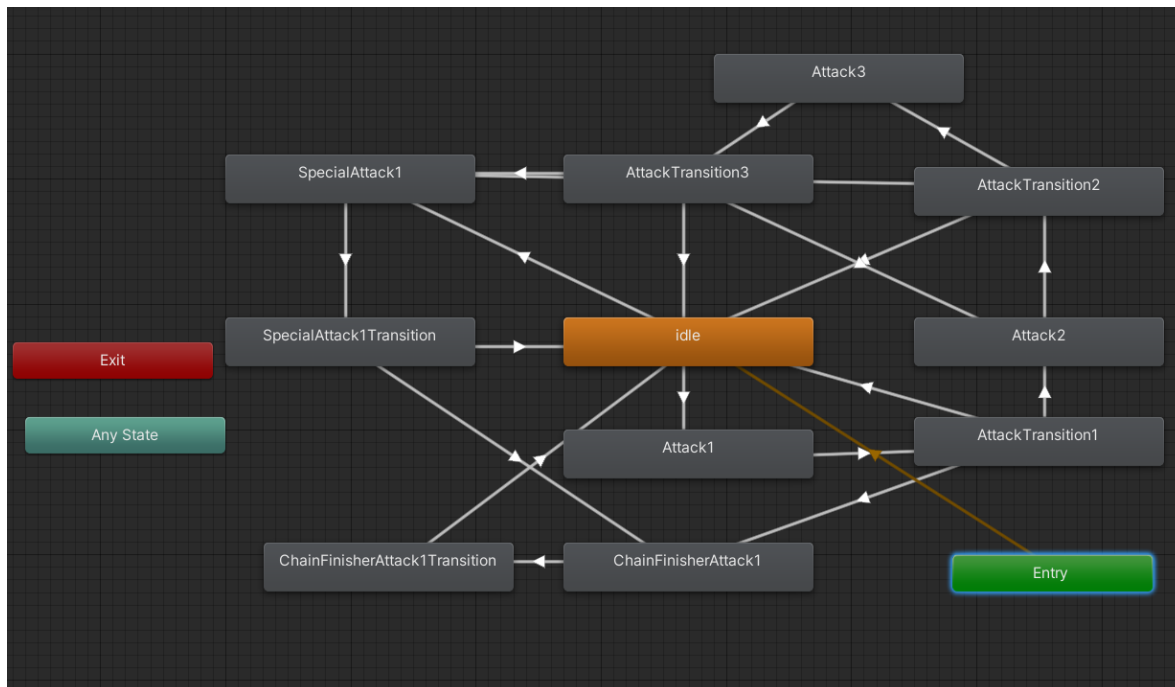
Animace

Zbraně

Každá zbraň se od sebe liší a má vlastní comba, a tak potřebuje vlastní animace. Ve hře máme celkem 3 hratelné zbraně: meč, kopí a nože.

Veškeré animace jsem tvořil v Unity pomocí okna Animation a poté je skládal v Animatoru. Každou zbraň jsem nastavil tak, aby na sebe jednotlivé útoky dobře navazovali a šlo tak kombinovat slabý i silný útok. Některé útoky jsou ze začátku hry zamčené a odemknou se v průběhu hry.

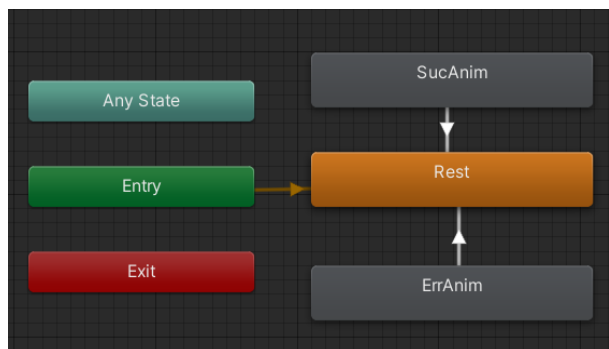
Ukázka animatoru zbraně:



UI

Všechny tlačítka by měla nějakým způsobem reagovat na kliknutí, aby bylo poznat, že fungují. Proto jsem ke každému tlačítku přidal vizuální a zvukový efekt, při kliknutí. Efekt se také může lišit podle toho, jestli akce proběhla úspěšně, či neúspěšně. Tlačítko je nejdříve v klidovém stavu, a po kliknutí se přehraje jedna z možných animací. Poté se tlačítko opět vrací do klidového stavu.

animátor tlačítka:



Ostatní

Ještě jsem do hry vytvořil animace přechodu mezi scénami, aby byl postup do dalšího levelu vizuálně plynulejší. A přidal jsem také animaci smrti hráče.

User Interface

Menu

V žádné dnešní hře nesmí chybět menu. Menu funguje jako rozcestník, odkud se hráč může dostat například do nastavení, nebo do hry. Při jejím vytváření jsem použil “Free Ui Pack” package z Unity Asset store. Ve hře jsou 2 různá menu: Main menu a Pause menu. Z Main menu lze ukončit program, přejít do nastavení nebo hrát. V Pause menu se jde místo ukončení programu, vrátit do Main menu.

Nastavení

Nastavení byla nejnáročnější část z celého UI. Obsahuje totiž jak tlačítka, tak i slidery pro ovládání hlasitosti zvuku, Dropdown na nastavení grafiky a rozlišení obrazovky a toggle pro ovládání Fullscreenu. Nastavení ještě obsahuje sub-menu pro nastavení ovládání. Všechny tyto komponenty odkazují na OptionsScript, který všechno zpracovává a nastavuje. Při jejím vytváření jsem opět použil “Free Ui Pack” package.

Options Menu:



Nastavení ovládání:



Inventář

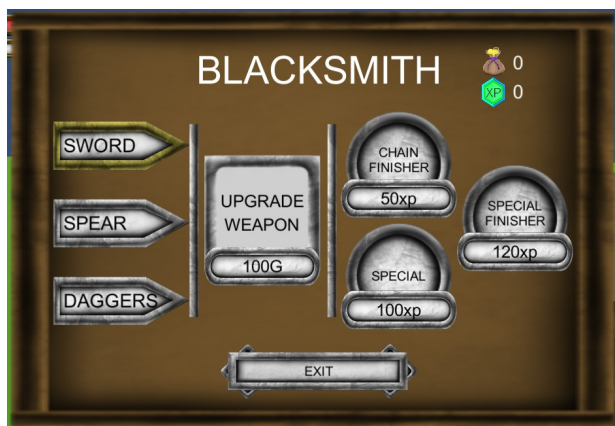
Je složený ze 3 sekce: Schopnosti, úkoly a vybavení. Na jeho vytvoření jsem použil jak “Free Ui Pack” package tak i vlastní Ui. Mezi sekcemi jde libovolně přepínat pomocí přidáných tlačítek. Do sekce s vybavením se ukládají všechny předměty a hráč si je pomocí myši může libovolně přesouvat mezi sloty, nebo se předmětem vybavit a poté ho použít.

Obchody

Do hry jsem vytvořil 4 různé obchody, a každý z nich je jedinečný. Nejjednodušší je stanice s vybavením, kde si hráč může pomocí tlačítek vybrat zbraň. Druhý v pořadí je Guilda, kam se ze scriptu vygeneruje seznam misí, mezi kterými pak jde přepínat. Další je kovář, u kterého si hráč může vylepšovat zbraně. Poslední a nejsložitější je

alchymista. Obsahuje Scrollbar s výběrem předmětů, které si hráč může prohlížet a kupovat. Na jejich tvorbu jsem opět použil “Free Ui Pack” package.

Ukázka kováře:



Ukázka alchymista:

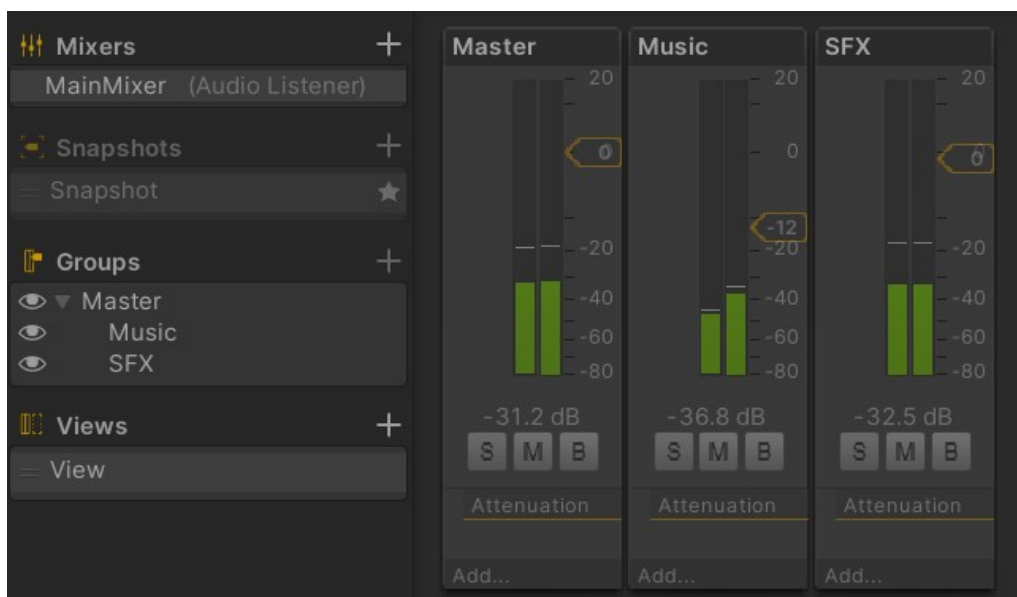


Zvuky

Audio Mixer

Protože ve hře je spousta různých zdrojů zvuku, které potřebujeme zvlášť ovládat, je nejlepší použít Audio Mixer. Zlepšíme tak přehlednost, rozšiřitelnost a konfigurovatelnost programu v oblasti audia. Audio Mixer umožňuje rozdělit si zvuky do skupin, a ty následně můžeme kombinovat a upravovat různými efekty. V mém projektu jsem si zvuky rozdělil na hudbu a efekty, které se kombinují do výstupu Master. Všem 3 skupinám jde tak nezávisle na sobě upravovat hlasitost.

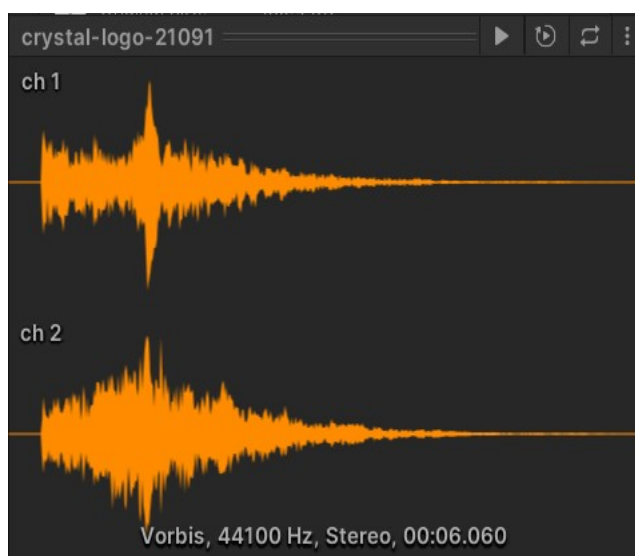
Audio Mixer - ukázka:



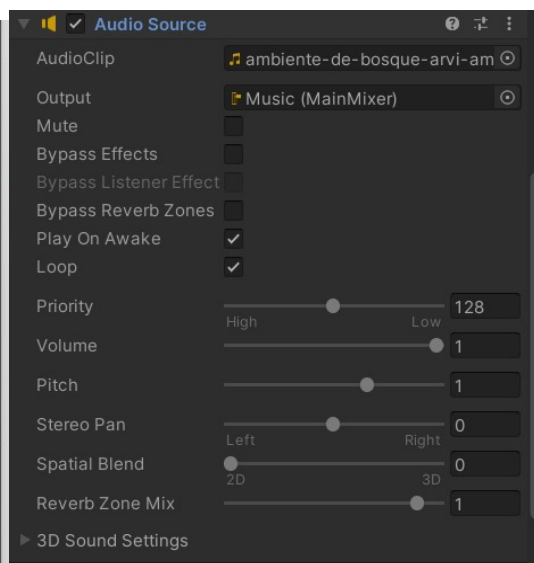
Hudba a zvukové efekty

Každá akce, kterou hráč udělá, by měla mít nějakou odezvu. Proto jsem do hry přidal mnoho zvukových efektů, které přidávají lepší pocit z hraní (např: při chůzi jsou slyšet zvuky kroků). V každé scéně také hraje hudba, ladící k prostředí (např: v lobby hraje klidná hudba).

ukázka zvukového efektu:



audio Source soundtracku:



Použité knihovny

JsonDotNet

Jelikož aktuální stav hry a její konfigurace je potřeba i po vypnutí programu zachovat, je nutné tyto data ukládat do souboru. Pro náš projekt jsem se rozhodl ukládat data ve formátu Json. A jelikož serializace a následné ukládání těchto dat může být dosti náročné, rozhodl jsem se pro zjednodušení práce použít knihovnu JsonDotNet.

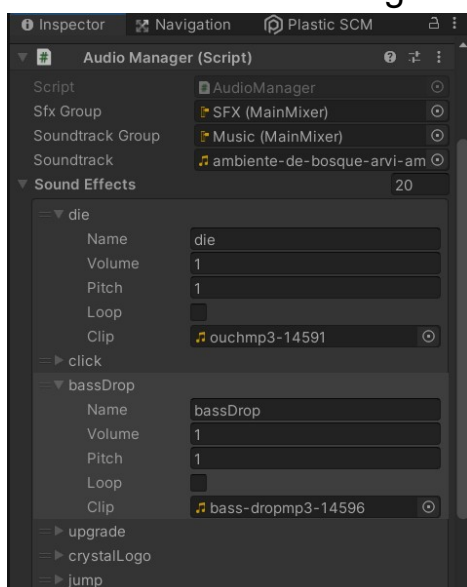
Ta mi umožňuje rychle a jednoduše načítat a ukládat třídy do Json souboru.

Scripty

Audio

Protože ve hře je mnoho různých zvukových efektů, tak jsem se pro větší přehlednost a jednodušší práci s audiem rozhodl napsat si vlastní Audio Manager script. A jelikož Audio Manager je ve hře vždy jen jeden použil jsem při tvorbě návrhový vzor singleton. Audio Manager je přístupný odkudkoli ve hře, takže z jakéhokoliv skriptu mohu přehrát jakýkoliv zvukový efekt. Pro zvukové efekty jsem si vytvořil vlastní třídu, u které jde nastavit audio klip, hlasitost, název efektu, Výšku tónu a zda se efekt má přehrávat ve smyčce.

Nastavení Audio Manageru: Metoda pro přehrání audio klipu:



```
//play an audio clip
Počet odkazů: 42
public void Play(string clipName)
{
    for (int i = 0; i < soundEffects.Length; i++)
    {
        MyAudio sfx = soundEffects[i];
        if (sfx.name == clipName)
        {
            sfx.Play();
            break;
        }
        if(i == soundEffects.Length - 1)
        {
            Debug.LogError("Sound " + clipName + " not found!");
        }
    }
}
```

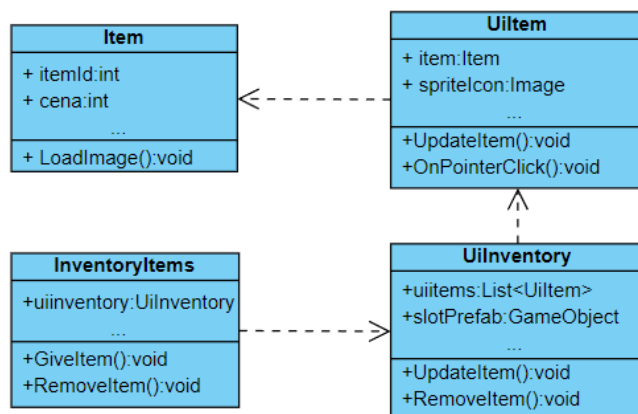
Inventář

Inventář je jeden z klíčových mechanik hry. Drží všechny zakoupené předměty, kterými se hráč může vybavit. Hráč má k dispozici několik slotů, mezi kterými si může předměty volně přesouvat za pomoci myši.

Inventář je rozdělený do 4 částí: Item, InventoryItems, UilInventory a UilItem. Item obsahuje informace o předmětu. UilItem zastupuje slot v inventáři a drží Item.

UilInventory pracuje se sloty v inventáři. A InventoryItems pracuje s celým inventářem a dokáže do něj přidávat nové předměty nebo je naopak odebírat.

Diagram inventáře:



Mise

Ve hře je také systém misí. Před vstupem do Dungeonu si hráč může vybrat jednu z několika zrovna nabízených misí. Během dalšího pokusu pak má cíl, za který dostane odměnu. Pokud se ale hráči nepovede cíl splnit, tak se mise zruší a hráč nedostane nic. Abych se vyhnul zbytečnému vypisování podmínek, naprogramoval jsem načítání misí z Json souboru. Každá mise má název, popis, odměnu a cíl (např. poraž 20 nepřátel). Dále jsem vytvořil panel v Inventáři, kde hráč může vidět aktuální misi a její postup, aby hra byla přehlednější a hráč si nemusel aktuální misi pamatovat.

Ukázka Json souboru s misemi:

```
{
  "missions": [
    {
      "type": "hunt",
      "title": "Slime hunt",
      "info": "in your next run kill 10 slimes",
      "enemy": "slime",
      "amount": 10,
      "price": 100
    },
    {
      "type": "hunt",
      "title": "berserk",
      "info": "in your next run kill 20 enemyes",
      "enemy": "any",
      "amount": 20,
      "price": 500
    }
  ]
}
```

Hráč

Hráč má na sobě jedny z nejdůležitějších scriptů, a to PlayerMovementScript a PlayerManagerScript a WeaponScript.

PlayerMovementScript:

Stará se o ovládání postavy a její pohyb. Jdou v něm upravit různé rychlosti hráče, výšku skoku, rychlost otáčení kamery, sílu gravitace a mnoho dalšího. Hlavní jsou metody HandleMovement a HandleCamera, které se starají o základní pohyb hráče.

Metoda HandleMovement:

```
Počet odkazů: 1
void HandleCamera()
{
    //getting mouse position input
    currRot.x += Input.GetAxis("Mouse X") * turnSpeed * Time.deltaTime;
    currRot.y += -Input.GetAxis("Mouse Y") * turnSpeed * Time.deltaTime;
    currRot.x = Mathf.Repeat(currRot.x, 360);
    currRot.y = Mathf.Clamp(currRot.y, minLookAngle, maxLookAngle);
    //rotating the camera
    gameObject.transform.rotation = Quaternion.Euler(0, currRot.x, 0);
    myCamera.transform.rotation = Quaternion.Euler(currRot.y, currRot.x, 0);
}
```

PlayerManagerScript:

Stará se o hráčovi životy, inventář a používání předmětů. Jde v něm nastavit maximální počet životů. Hlídá jestli je hráč naživu a při smrti přehraje animaci a vrátí hráče do lobby.

Ubírání životů v PlayerManagerScriptu:

```
//Player takes damage and potentially dies
Počet odkazů: 6
public void TakeHit(float damage)
{
    //take damage
    health -= damage;

    //if health is below 0, die
    if(health <= 0)
    {
        PlayerMovement PM = gameObject.GetComponent<PlayerMovement>();
        PM.enabled = false;
        WeaponScript WS = gameObject.GetComponentInChildren<WeaponScript>();
        WS.enabled = false;
        anim.enabled = true;
    }

    //set the health slider
    healthSlider.value = health;
}
```

WeaponScript:

Stará se o zbraň a její používání. Dají se v něm nastavit hitboxy a poškození jednotlivých útoků. Hlavní je metoda Attack, která kontroluje zda zbraň něco trefila. Pokud je trefený objekt nepřítel udělí mu nastavené poškození a vytvoří particle effect.

Zásah nepřítel ve WeaponScriptu:

```
//if its enemy, hit him
if (col.CompareTag("Enemy") || col.CompareTag("Boss"))
{
    if (!col.isTrigger)
    {
        EnemyAi enemy = col.gameObject.GetComponent<EnemyAi>();
        enemy.TakeDamage(damage);
        Vector3 dir = (enemy.transform.position - cam.transform.position).normalized;
        enemy.ApplyForce(dir * knockback);

        Vector3 pos = cam.transform.position + offset * cam.transform.forward;
        Vector3 eul = new Vector3(cam.transform.eulerAngles.x, cam.transform.eulerAngles.y - 90, cam.transform.eulerAngles.z);
        Quaternion rot = Quaternion.Euler(eul);
        GameObject particle = GameObject.Instantiate(damageParticle, pos, rot);
        GameObject.Destroy(particle, 1);
    }
}
```

Obchody

Obchody jsou klíčovým prvkem hry. Nachází se v Lobby a právě u nich si může hráč kupovat předměty, vylepšovat vybavení a připravovat se na další pokus. Ve hře jsou celkem 4 obchody

BlacksmithScript:

Slouží k vylepšování zbraní a odemykání nových bojových technik.

Pro odemknutí bojové techniky potřebuje hráč dostatek peněz a zkušeností s danou zbraní. Na vylepšení zbraně mu stačí peníze.

AlchemistScript:

U alchymisty se dají kupovat předměty, které hráč může použít v dungeonu. Každý předmět má svoji cenu a pokud hráč nemá dost peněz, nemůže si předmět koupit. Zakoupené předměty se hráči automaticky přidávají do inventáře a může je pak libovolně používat dokud mu nedojdou.

GuildScript:

Slouží k přijímání misí. Nejprve mise načte ze souboru a poté z nich náhodně vybere 3 a dá je hráči do nabídky. Hráč si nemůže vybrat víc misí na jednou, ale před vstupem do dungeonu si jí ještě může vyměnit.

EquipScript:

Hráč si u stanice s vybavením může vybrat jednu ze 3 zbraní, s kterou půjde do dungeonu.

Vylepšování zbraně v BlacksmithScriptu:

```
//upgrades selected weapon
Počet odkazů: 0
public void Upgrade(int weaponIndex)
{
    if (GameData.weapons[weaponIndex].level < GameData.weapons[weaponIndex].upgradeCosts.Length)
    {
        if (GameData.weapons[weaponIndex].upgradeCosts[GameData.weapons[weaponIndex].level] <= GameData.money)
        {
            GameData.money -= GameData.weapons[weaponIndex].upgradeCosts[GameData.weapons[weaponIndex].level];
            GameData.weapons[weaponIndex].level++;
            AudioManager.instance.Play("upgrede");
        }
        else
        {
            AudioManager.instance.Play("error");
        }
    }

    updateStatus();
    GameData.SaveGameData();
}
```

kupování předmětu v AlchymistScriptu:

```
if (GameData.money >= item.cena)
{
    GameData.money -= item.cena;
    GameObject controller = GameObject.FindGameObjectWithTag("GameController");
    InventoryItems myInventory = controller.GetComponent<InventoryItems>();
    myInventory.GiveItem(selectedItemIndex);
    buyButtonAnim.Play("Base Layer.SucAnim", 0);
    AudioManager.instance.Play("upgrade");
}
else
{
    buyButtonAnim.Play("Base Layer.ErrAnim", 0);
    AudioManager.instance.Play("error");
}

moneyTxt.text = GameData.money.ToString();
GameData.SaveGameData();
```


Ostatní

SkillUnlock.cs:

Slouží k odemykání schopností a stará se o UI schopností.

Nepřátelé, menu, pasti, ukládání hry a GameManager dělal pan Růžička.

Animace nepřátel dělal pan Růžička.

Jednotlivé levely dělal pan Růžička.

Zdroje:

Během práce jsem čerpal z několika zdrojů

řešení problémů:

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

dokumentace:

- Unity: <https://docs.unity3d.com/Manual/index.html>
- Visual Studio: <https://docs.microsoft.com/cs-cz/visualstudio/windows/?view=vs-2022>

3D modely:

- <https://free3d.com/>

Hudba a zvukové efekty:

- <https://mixkit.co/free-sound-effects/>
- <https://pixabay.com/sound-effects/>

závěr

Prace na tomto projektu pro mě měla velký přínos. Toto byl můj první velký projekt v Unity a naučil jsem se díky němu mnoho věcí o vývoji her. Během vývoje jsem zjistil, že vývoj her je časově mnohem náročnější, než jsem si původně myslel. Přiznám se, že jsem nestihl všechno, co jsem původně plánoval. Například zabudovat do hry magii a rozšířit počet zbraní, schopností a předmětů. Na hře ale hodlám dále pracovat dokud nedosáhne dostatečné úrovně a doufám že se jednoho dne dočká plnohodnotného vydání