



STŘEDNÍ ŠKOLA PRŮMYSLOVÁ
A UMĚLECKÁ, OPAVA

ZÁVĚREČNÁ STUDIJNÍ PRÁCE

dokumentace

Videoherní demo

Michal Ondráček

OPERATION
WARHAMMER

Obor: 18-20-M/01 INFORMAČNÍ TECHNOLOGIE
se zaměřením na počítačové sítě a programování

Třída: IT4

Školní rok: 2019/2020

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité informační zdroje.

Souhlasím, aby tato studijní práce byla použita k výukovým účelům na Střední průmyslové a umělecké škole v Opavě, Praskova 399/8.

V Opavě 31. 12. 2020

podpis autora práce

ANOTACE

Cílem projektu bylo vytvoření hratelné postavy a prostředí, ve kterém se bude nacházet. Hlavní postava je schopna pohybu ovládaného hráčem, také střelby, přebíjení a výměny zbraní. Toho bylo dosaženo za pomoci herního enginu Unreal Engine 4 a jazyka C++. Pro tvorbu 3D modelů byl použit 3D modelovací software 3ds Max, na texturování modelů programy Gimp 2.10 a Substance Painter 2. V neposlední řadě se našlo využití pro software Audacity k nahrávání a úpravě zvuku.

ANNOTATION

The main goal of this project was to create playable character and the enviroment, in which this chracter will be placed. Main chracter is able to move, shoot, reload and switch weapons on users command. This was accomplished with the help of game engine Unreal Engine 4 and C++ programming language. For creating 3D models have benn used 3D modelling software 3ds Max, for texturing programs such as Gimp 2.10 and Substance Painter 2. Last but not least been used software Audacity for audio recording and editing.

Klíčová slova

C++, video hra, 3D modelování, Unreal Engine 4

OBSAH

OBSAH.....	3
ÚVOD	5
1 ENGINE, GRAFIKA A ZVUK.....	6
2 VYUŽITÉ TECHNOLOGIE	7
2.1 GRAFIKA	7
2.1.1 3DS MAX	7
2.1.2 SUBSTANCE PAINTER 2.....	7
2.1.3 GIMP 2.10	7
2.1.4 BLENDER 2.8.....	7
2.2 ENGINE.....	8
2.2.1 UNREAL ENGINE 4.....	8
2.2.2 VISUAL STUDIO 2017	8
2.2.3 C++	8
2.2.4 SOURCETREE.....	8
2.3 AUDIO	9
2.3.1 AUDACITY	9
2.3.2 MIKROFON.....	9
3 ZPŮSOBY ŘEŠENÍ A POUŽITÉ POSTUPY.....	10
3.1 TVORBA 3D MODELU	10
3.1.1 MODELOVÁNÍ	10
3.1.2 TEXTURY	13
3.1.3 ANIMACE.....	15
3.2 TVORBA HLAVNÍ POSTAVY.....	17
3.2.1 VZHLED POSTAVY	17
3.2.2 KÓD A PROTOTYPOVÁNÍ	17
3.2.3 POHYB.....	18
3.3 TVORBA UI.....	18

3.3.1	HLAVNÍ MENU.....	18
3.3.2	POZASTAVENÍ HRY	19
3.3.3	UI HRÁČE.....	20
3.4	TVORBA AUDIO EFEKTU.....	20
3.5	TVORBA HERNÍHO PROSTŘEDÍ	21
4	POSTAVA HRÁČE	23
5	HERNÍ PROSTŘEDNÍ.....	24
	ZÁVĚR	25
	SEZNAM POUŽITÝCH INFORMAČNÍCH ZDROJŮ	26
	SEZNAM PŘÍLOH	27

ÚVOD

Každý člověk zná koncept video hry. Je to způsob zábavy, která může sloužit čistě k odreagování, ale i ke sdělení hlubší myšlenky. Jelikož je tato škála vcelku rozsáhlá a stále vznikají nové a nové herní koncepty, rozhodl jsem se také nahlédnout do problematiky vytváření her. Myšlenkou vytvořit plně funkční hru jsem se zabýval již dříve, avšak tyto pokusy byly úspěšné jen částečně. Proto je má práce zaměřena na zhotovení funkční demo verze hry. Pokud jsem schopný hry hrát, proč také jeden exemplář nevytvořit? Abych nastínil problematiku projektu, je třeba si uvědomit, kolik věcí bereme ve hrách za samozřejmost. Uvedu příklad. Postava hráče musí mít naprogramované nějaké základní chování, je potřeba virtuální kamera (kterou hráč uvidí svět kolem sebe), poté se musíme postarat o to, aby měl hráč pocit, že je opravdu ve hře, tzn. tělo, a to už je oříšek. Vytvoření těla má několik fází. První fáze je koncept. V této části se musíme rozhodnout, jak bude postava vypadat, případně jí zvolíme minulost pro větší uvěřitelnost. Jelikož by se koncept dal přirovnat spíše k nákresu, je v další fázi třeba dotáhnout detaily postavy a začít s modelováním v libovolném 3D programu. Další fázi jsou textury hrdiny, tedy jakýsi barevný potah modelu. Poté přichází na řadu animace a v konečné fázi se můžeme soustředit na otázku audia. Samozřejmě tyto fáze mají své podfáze, podfáze podfází apod. A tento proces vyžaduje každý objekt vsazený do hry. Mým cílem bylo tedy vytvořit ukázkou hry (tzv. demo) od úplného začátku.

1 ENGINE, GRAFIKA A ZVUK

V oblasti vývoje her nejsem začátečník, jak jsem se již zmínil, není to můj první pokus o vytvoření hry. S touto problematikou jsem tedy seznámen. Nejprve je třeba zvolit prostředí, ve kterém se bude hra vyvíjet. Pro potřeby projektu bohatě postačí herní engine Unreal Engine 4 a IDE Visual Studio 2017. Protože bude projekt verzován, grafická aplikace Sourcetree, která nabízí přehledné prostředí pro verzování kódu i jiných souborů, je nejsnadnější možností jak toho docílit. Dále si musíme opatřit software pro tvorbu modelů. V tomto případě používám 3ds Max 2020, s kterým jsem se seznámil v rámci předmětu počítačové grafiky ve třetím ročníku a stále jej využívám. Jde totiž o profesionální program a průmyslový standart. Zpracování textur je také součástí procesu tvoření her. Pro texturování modelů použiji Substance Painter 2, protože jde o profesionální program, a i přesto je studentům dostupný zdarma. Gimp 2.10 budu používat místo Photoshopu CC od firmy Adobe, pro základní úpravu textur. Jelikož mám v úmyslu postavu hráče animovat použiji víceúčelový program Blender 2.8 na tvorbu animací. Jde totiž o program, se kterým mám již zkušenosti.

2 VYUŽITÉ TECHNOLOGIE

2.1 Grafika

2.1.1 3ds Max

Jde o software pro 3D modelování, ve kterém jsem vytvářel každý model vsazený do herního prostředí. Pokud má uživatel zkušenosti s jinými programy pro modelování, rychle se v něm zorientuje.

2.1.2 Substance Painter 2

Tento 3D grafický program je určen pro tvoření textur na zhotovených 3D modelech. Je to jeden z nejlepších a uživatelsky příjemných programů na trhu. Často se využívá ve spolupráci se Substance Designer, který jsem ovšem v projektu nepoužil.

2.1.3 Gimp 2.10

Grafický software pro úpravu rastrových obrázků. V mé práci byl využit pro finální úpravy textur, loga apod. Kvůli absenci Adobe Photoshop na mém počítači jsem zvolil tohoto dostačujícího konkurenta.

2.1.4 Blender 2.8

Také jsem využil všestranný program Blender. Jde o software pro modelování, animaci, úpravu videa a zároveň základní herní engine. Ve své práci jsem použil jeho schopnosti pro animaci hlavní postavy. Důvodem výběru byla má předchozí znalost programu, a fakt že opravdu jde o nejlepší profesionální program s nulovou cenovkou.

2.2 Engine

2.2.1 Unreal Engine 4

Zkráceně UE4. Jedná se o herní engine společnosti Epic Games, vyvíjený od roku 2014, který je veřejnosti přístupný zcela zdarma a bez omezení. Používá se k tvorbě fotorealistických i stylizovaných scén, přičemž u renderování fotorealistických scén není přehnaně náročný na hardware. Kódování probíhá pomocí externího IDE v C++. Jednou z výhod je také možnost přímého napojení source control.

2.2.2 Visual studio 2017

Externí vývojové prostředí pro UE4. Vytvořené společností Microsoft. Jde o IDE jako každé jiné, ovšem narozdíl od jiných má několik užitečných funkcí pro vývoj aplikací, jako např. TODO list. Důvodem výběru byla kompatibilita s UE4, jednoduchá orientace v programu a jeho funkcích.

2.2.3 C++

Objektově orientovaný programovací jazyk vycházející z jazyka C. Jeden z nejrozšířenějších programovacích jazyků. V projektu je použit kvůli oficiální podpoře jazyka Unreal Enginem.

2.2.4 Sourcetree

Je vizuální aplikace pro GIT verzování kódu. Je kompatibilní s UE4 nabízí napojení na GitHub či Bitbucket, také možné využití LFS na těchto platformách. Jde o přehlednější verzování kódu než za pomoci příkazového řádku.

2.3 Audio

2.3.1 Audacity

Jde o jednoduchý program pro nahrávání a úpravu zvuku. Uživatel je schopen např. pracovat s několika stopami zvuku najednou, audio lze modulovat dle potřeby a výstupní soubor lze samozřejmě „označkovat“ názvem, popisem, apod.

2.3.2 Mikrofon

Samotný program pro nahrávání zvuku je bez mikrofону zbytečný. V projektu byl použit mikrofon Auna 900. Jde o studiový mikrofon, obohacený o pop filtr. Důvodem volby byla přijatelná cena a jeho vlastnosti, které se při správných podmínkách blíží studiové kvalitě nahrávání zvuku.

3 ZPŮSOBY ŘEŠENÍ A POUŽITÉ POSTUPY

3.1 Tvorba 3D modelu

3.1.1 Modelování

První podstatnou věcí, na kterou je třeba se soustředit, nejen u tvoření modelu, je koncept. V mém případě jsem použil pro koncepty obyčejnou tužku a papír, popř. tablet (ten je výhodnější, jelikož není nutná následná digitalizace obrázků). Jeden z konceptů můžete vidět na obrázku č. 1 níže.



obrázek č.1

Pokud už máme hrubou představu toho, jak bude model vypadat, můžeme začít s modelováním. V této fázi postupně přidáváme další detaily a bereme si inspiraci z referenčních obrázků, pokud nějaké máme k dispozici. Na obrázku č. 2 , také č. 3 jsou k vidění reference pro hodinky hlavní postavy.

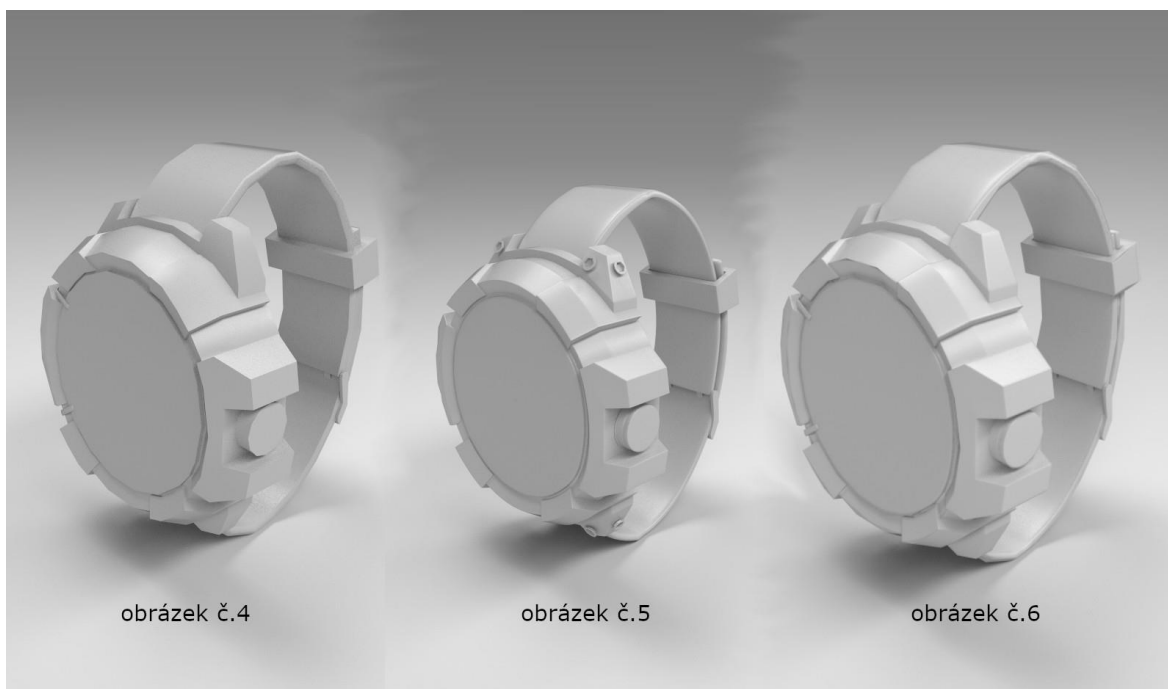


Obrázek č.2



Obrázek č.3

Pro maximální kvalitu modelu je nutné vytvořit tzv. objekt s vysokými detaily (high poly = hp) a objekt s nízkými detaily (low poly = lp), na který následně “natiskneme” vysoké detaily hp objektu v texturovacím softwaru. Výsledek můžete vidět na obrázku č.6 , tzv. hp model se nachází na obrázku č. 5, lp model můžeme vidět na obrázku č. 4.

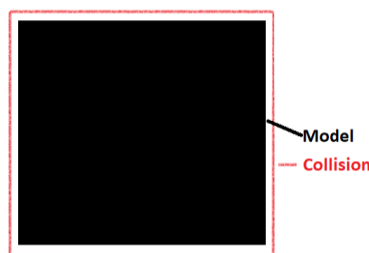


obrázek č.4

obrázek č.5

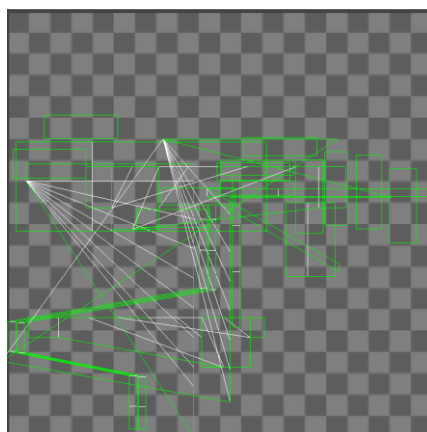
obrázek č.6

Dalším krokem je vytvořit tzv. kolizi objektu (anglicky “collision”). Kolize zajišťuje, že při nějaké akci, např. sražení dvou těles, se tělesa opravdu srazí a nebudou pokračovat ve svém dosavadním pohybu skrz to druhé. Vlastní kolize objektu se tvoří v modelovacím programu např. zkopírováním modelu, odstraněním několika polygonů a doplnění názvu modelu o předponu UCX_ . Pro lepší pochopení můžete vidět na obrázku č. 7 pomyslný model a jeho kolizi, která jej ohraničuje.

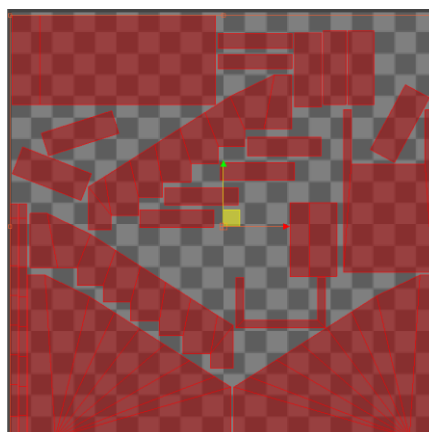


obrázek č.7

Posledním krokem před exportováním modelu je správné rozložení UV mapy. Tato mapa je důležitá pro pozdější proces texturování modelu v texturovacím program. Jedná se vlastně o rozložení ploch 3D objektu do 2D prostoru, čímž získáme výsledek podobající se síti tělesa, jaký známe například z hodin geometrie na základní škole. Pokud bychom tento krok jednoduše přeskočili, vypadala by UV mapa jako na obrázku č. 7. Správné rozložení mapy je k vidění na obrázku č.8. Na přiložených obrázcích můžeme rozpoznat, že pokud se jednotlivé plochy objektu překrývají, nevypadá to správně a později nedosáhneme správného zobrazení textur. Zelené čáry jsou hranami ploch.



obrázek č.7

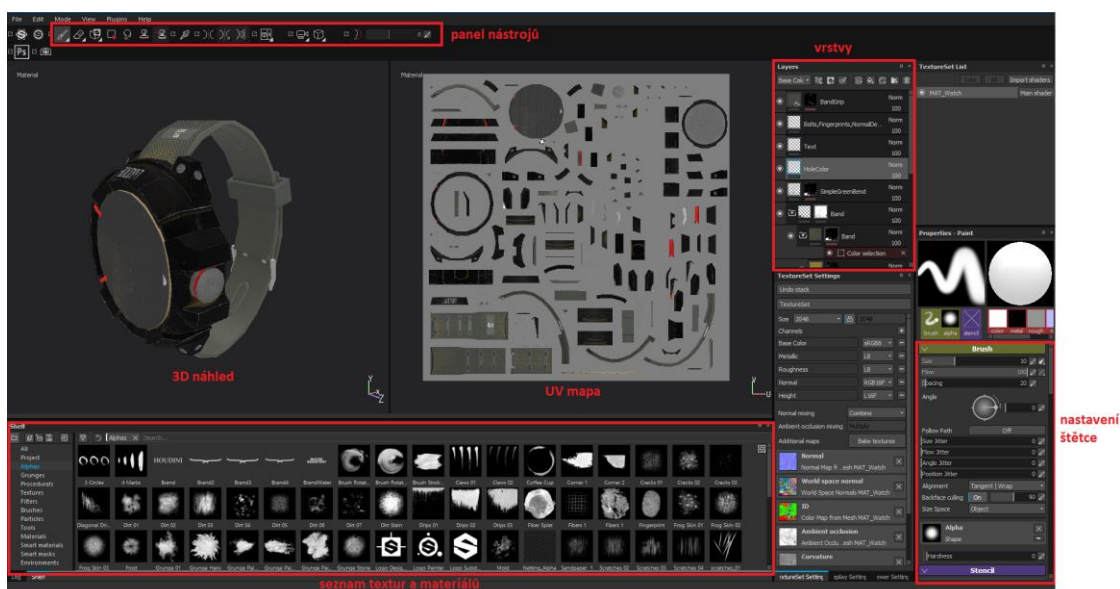


obrázek č.8

Pro rozložení UV mapy používáme nástroj (za předpokladu že pracujeme v 3ds Max) nazvaný Quick Peel, popřípadě nástroj Flatten by Smoothing Group. Tyto nástroje nám automaticky rozloží UV mapu, avšak je nutná korekce. Je potřeba, aby každá hrana 3D modelu, jejíž polochy mají vůči sobě sklon více než 45° , byla dělicí hranou pro tyto plochy. K ručnímu oddělení používáme nástroj Break. Pokud je naopak potřeba tzv. sešít dvě plochy dohromady, použijeme nástroje Stitch (Stitch to Target, Stitch Average, apod.). Teprve až dosáhneme výsledku, který je srovnatelný s obrázkem č. 8 můžeme začít model exportovat. Hp a lp model můžeme exportovat ve formátech, které podporuje herní engine a texturovací software, tedy většinou FBX nebo OBJ.

3.1.2 Textury

Druhou fází ve tvoření 3D modelu (v rámci mého projektu) bylo, dalo by se říci, nabarvení vymodelovaného objektu. Jelikož jsme si již ve fázi modelování připravili své malířské plátno (UV mapu), můžeme se nyní soustředit na obarvení, tzv. texturování modelu. Proces je podobný klasickému kreslení na papír. Pomocí štětce nanášíme po vrstvách barvy na model. Tyto vrstvy, označované anglicky jako layers, jsou srovnatelné s vrstvami v rastrových, či vektorových programech pro kreslení. Na obrázku č. 9 můžete vidět ukázkou uživatelského prostředí texturovacího programu Substance Painter 2, který byl v projektu použit.



obrázek č.9

V panelu nástrojů si můžeme zvolit, jak budeme texturu upravovat, máme k dispozici štětec, gumu, výběr polygonů, nástroj zrcadla apod. Texturu můžeme upravovat na dvou různých místech, jednak v okně 3D náhledu, nebo v okně s UV mapou, osobně preferuji první možnost. Dále v panelu nastavení štětce zvolíme jeho vlastnosti. Vlastnosti jako barvu a její průhlednost (opacity), hrubost jejího povrchu (roughness), výšku povrchu (height), lesk povrchu (metallic) a normal map. Součet těchto základních vlastností nazýváme materiálem. Samozřejmostí jsou pak velikost, úhel, tlak, náhodnost úhlu a pozice štětce při kreslení. V neposlední řadě máme k dispozici seznam importovaných texture a materiálů. Jelikož ne vždy je jednodušší vytvořit texturu povrchu digitálně, je v

jistých případech lepší použít reálné snímky z fotoaparátu. U takovýchto případů máme na výběr, jak snímky získáme. První možností je internet. Existuje mnoho stránek, které nabízejí po zaplacení jistého obnosu peněz, stažení kvalitních textur pro herní vývoj, ale i reklamní či neziskové účely. Jsou ovšem i takové stránky, které nabízejí své základní služby zdarma, a pokud je s nimi uživatel spokojen, může si připlatit za prémiové funkce. Příkladem je stránka [texture.com](https://www.texture.com), kterou jsem použil. Druhou možností je tvoření vlastních snímků. Za pomoci fotoaparátu a při správném nasvětlení, získáme snímek, jež následně upravíme v rastrovém editoru obrázků. Když máme připravené textury pro materiál, poslední co musíme udělat je pouze nahlízet do referenčních obrázků a postupně na model materiál nanést. Výsledek texturování můžeme vidět na obrázku č. 10 a obrázku č. 11.



obrázek č.10

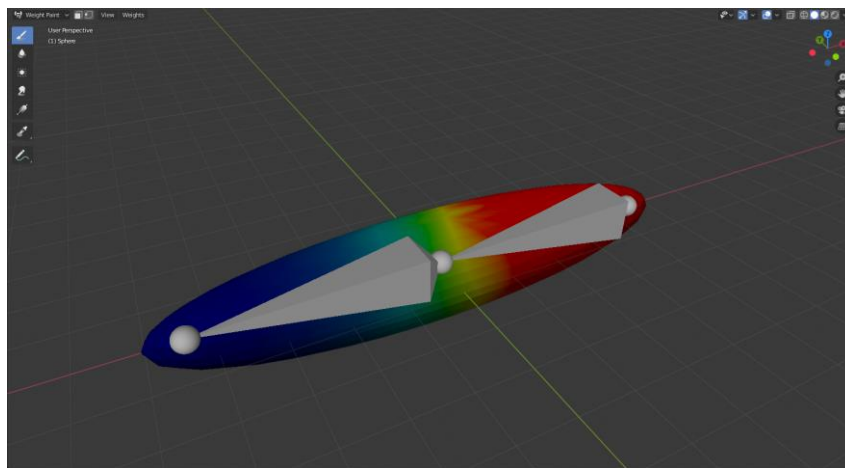


obrázek č.11

3.1.3 Animace

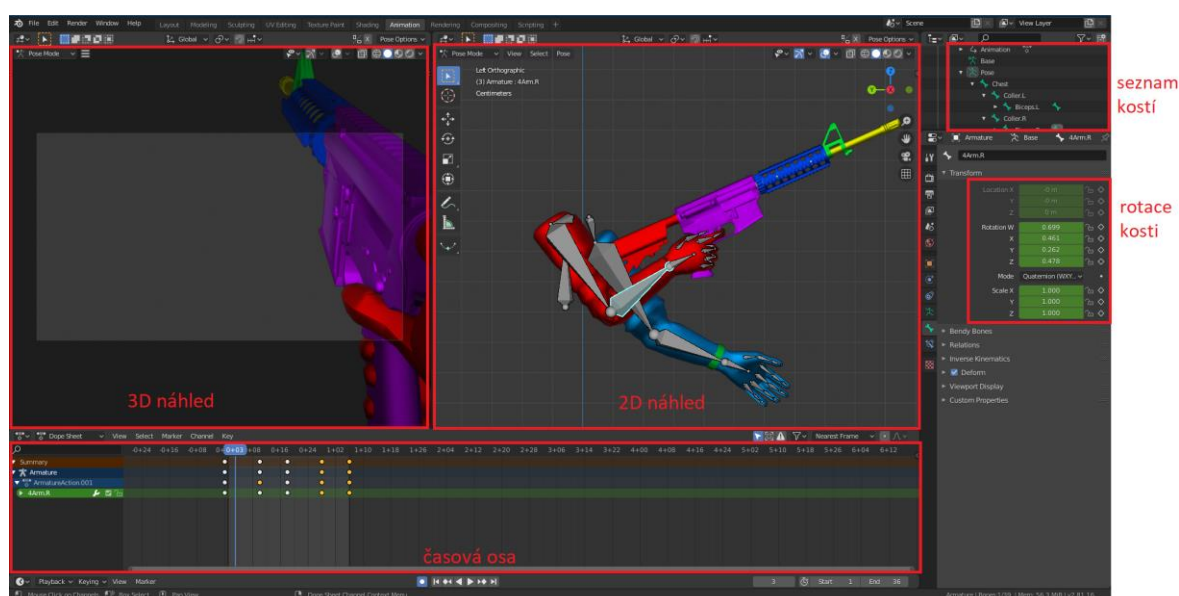
Tento krok není povinností u všech modelů. Animaci používáme pouze u modelů jako jsou postavy, objekty, s kterými postavy provádějí interakci, či látka, kterou ovlivňuje poryv větru, apod. V tomto projektu byla použita pouze animace hlavní postavy. Byly vytvořeny animace pro chůzi, skok, přebíjení a střelbu. Proces tvorby animace je zdlouhavá záležitost. Animátor si nejdříve pořídí předlohu, podle které může rozpohybovat model, tak aby byl pohyb co nejpřirozenější. Prvním krokem je tedy např. opatření videa, kde je pohyb zaznamenán. Na záznamu je viditelný pohyb, který referenční objekt vykonává, a na časové ose vlastně také rychlost jakou ho vykonává. To pro nás znamená, že pohyb dokážeme replikovat. Máme totiž bod, ve kterém se bude referenční objekt nacházet a také moment, ve kterém se tam bude nacházet. Pokud máme zvolený model, který budeme animovat, referenční video a program pro tvorbu animace, začneme s přípravou modelu.

Začneme „kostmi“ modelu, procesu přidávání kostí a oblastí, na které budou kosti působit se říká rigging. Kostí nejsou jen pro modely postav, kosti a kostra modelu jsou totiž nositeli animace. To znamená, že i pro animaci pohybu závěru zbraně je nutné závěr napojit na kost, která jím bude pohybovat. Na obrázku č. 12 je možné vidět proces přiřazování kostí k modelu. Modrá barva znamená, že kost nebude ovlivňovat oblast vůbec. Červená barva značí maximální ovlivnění oblasti zvolenou kostí. K tomuto přiřazování hodnot používáme nástroj v programu Blender označený jako Weight Paint. Proces je tedy známý pod výrazem „weight painting“.



obrázek č.12

Jakmile máme model správně nastaven, můžeme se zaměřit na samotnou animaci. Za použití referenčních snímků nejdříve dáme modelu (pro lepší přehlednost v tomto případě lidské postavě) základní pohyb, stačí naprosto jednoduchý a přímočarý. Poté, co jsme spokojeni se svým základem, se budeme postupně propracovávat do větších detailů pohybu. Např. při dopadu člověk tlumí náraz určitým pohybem nohou, tento pohyb vyvažuje pažemi a horní částí těla. Všechny drobné detaily pohybu těchto různých částí těla musíme zachytit, abychom získali věrohodnou animaci pohybu člověka. Na obrázku č. 13 je viditelný proces animace rukou hlavního hrdiny a prostředí programu.



obrázek č.13

Prostředí je pro uživatele poměrně přehledné. K dispozici máme dva náhledy, 2D a 3D, ve kterých upravujeme pohyb modelu. Na časové ose máme vyznačenou kost kterou hýbeme, a čas ve kterém dojde ke změně pohybu. Dále máme k dispozici seznam všech kostí patřících k modelu. Seznam má vlastní vyhledávač, takže není problém najít kosti i podle názvu. Poslední ze základních oken v programu je to s tabulkou velikosti, rotace a lokace kosti. Jelikož jsou kosti vzájemně spojeny, je možnost lokace nedostupná. Kosti se totiž nikam nepřesouvají, pouze rotují, stejně jako v lidském těle.

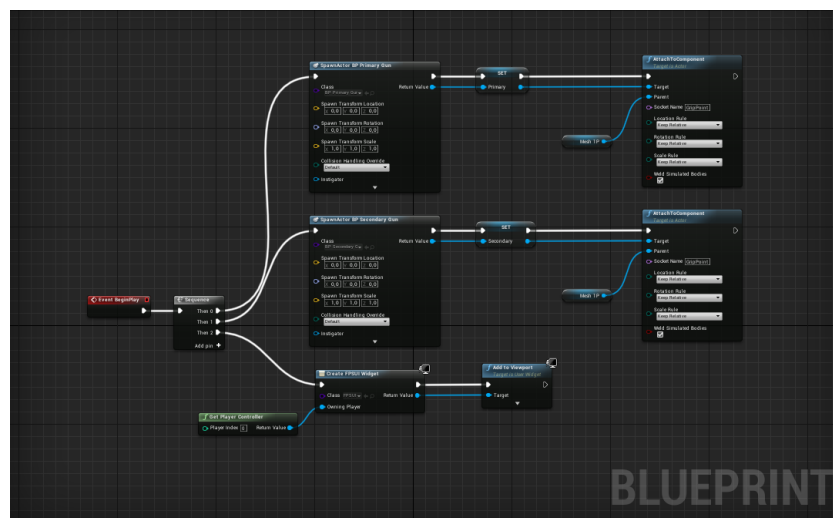
3.2 Tvorba hlavní postavy

3.2.1 Vzhled postavy

Postup jakým se tvoří 3D model postavy jsem již objasnil. Když máme připravený model i s animacemi, importujeme vše do UE4. Tady vytvoříme novou třídu např. s názvem Character. Z této třídy pak „child“ třídu Player, kterou budeme upravovat. Základním objektem je kamera, kterou bude hráč pozorovat svět. Kamera má několik možností nastavení od barevnosti filtru, přes úhel pohledu, až po fyziku a aktivaci. Aby se ruce hýbaly s kamerou a byly stále hráči na očích, je model rukou podřazený kameře. Tedy pokud se hýbe kamera, ruce jsou k ní „přibité“ a zůstávají zabírané kamerou. V mém projektu bylo zapotřebí ještě při každém spuštění hry vložit do rukou postavy zbraň. Toho bylo docíleno kódem a modelem zbraně, který je samostatnou třídou.

3.2.2 Kód a prototypování

V rámci kódování jsem pro prototypování využil integrované vývojové prostředí UE4. Tento engine totiž obsahuje možnost vlastního vizuálního programování. V tomto případě je slovem vizuální myšleno, že uživatel programuje pomocí tabulek, které spojuje dohromady, přičemž každá tabulka znázorňuje programovací funkci. Ukázka vývojového prostředí je na obrázku č. 14.



obrázek č.14

V první řadě jsem tedy vytvořil prototyp kódu v tomto prostředí a následně jej přepsal do C++ přímo v IDE Visual Studio. Tento postup ovšem nebyl vždy potřeba, v některých případech bylo jednodušší napsat přímo kód. Snadné porozumění vývojovému prostředí UE4 naneštěstí přináší i fakt, že většina internetových tutoriálů a fór se spíše zabývá vizuálním programováním, než programováním v C++. Tudíž při vývoji jsem musel často převádět na kód již hotová řešení z vizuálního programování.

U tvoření hlavní postavy jsem nejdříve vytvořil „parent“ třídu Character z níž pak byla vytvořena „child“ třída Player. Třída Character má v sobě základní funkce, tzn. že detailnější podoba, jako vzhled rukou, počet životů, animace, mohu upřesnit v třídě Player. Výhodou tohoto postupu je fakt, že mohu vytvořit různé „child“ třídy a nebude to mít vliv na „parent“ třídu Character. Kód postavy obsahuje především funkce pro střelbu, přebíjení a pohyb.

3.2.3 Pohyb

K pohybu hlavní postavy byl použit kód, který vlastně jen říká, jak a kdy se postava hýbe po ose X,Y. Například po ose X se postava pohybuje rychlostí určenou proměnnou „speed“ nebo „velocity“. A námi vytvořené animační schéma (AnimationBlueprint) pak z této proměnné určí, zda se má přehrávat animace stojícího nebo běžícího hráče.

3.3 Tvorba UI

UI tedy User Interface je samozřejmostí každé hry. V projektu jsou obsaženy tři nejčastější typy uživatelského prostředí.

3.3.1 Hlavní menu

Hlavní menu her může být pouhý obrázek s ikonkami start a exit, ale také se může jednat o místo k nastavení zobrazení hry, anebo k odkázání na autory hry. Pro vytvoření hlavního menu je zapotřebí vytvořit také nový herní „level“, ze kterého budeme spouštět jiné hratelné úrovně hry. Protože pozadím hlavního menu bude 3D scéna, je potřeba nastavit světlo, stíny a importovat objekty. Poté vytvoříme UI.

UE4 nabízí přehledné prostředí pro úpravu UI, proto nastavení funkcí tlačítek jako start, exit, options, apod. nezabere tolik času. Tlačítko start odkazuje na funkci načítání herní mapy, předtím vymaže aktuální UI z obrazovky a nahradí ho načítací obrazovkou. Exit odkazuje na funkci pro bezpečné ukončení hry. Po kliknutí na tlačítko options, či credits se smaže UI hlavního menu a načte se nové UI pro příslušné tlačítko. Výsledkem je funkční hlavní menu na obrázku č. 15.



obrázek č. 15

3.3.2 Pozastavení hry

K pozastavení hry je zapotřebí zvolit klávesu, která bude pozastavení spouštět. Zvolíme nejčastější klávesy pro tento účel, a to ESCAPE a P. V nastavení enginu pojmenujeme tyto klávesy jako Pause. Podobným způsobem jako hlavní menu vytvoříme pause menu UI. Bude mít tři tlačítka. Tlačítko play nám opět spustí hru, Main menu nás přesměruje zpět do hlavního menu a Exit ukončí hru. Po stisku klávesy P nebo ESC se stávající UI hráče smaže a nahradí jej pause menu UI. Pro pozastavení hry využijeme již vytvořenou funkci Pause Game, která je součástí enginu. Finální menu pozastavení hry můžete vidět na obrázku č. 16.



obrázek č. 16

3.3.3 UI hráče

Hráč bude mít k dispozici přehled o své munici v zásobníku zbraně a počtu munice, kterou si nese. Tohoto docílíme pomocí jednoduchého UI, ve kterém přidáme do levého dolního rohu dva texty. Jeden pro munici v zásobníku a jeden pro munici, kterou hráč ponese s sebou. Tento text se bude po vystřelení náboje updatovat z proměnné „munice ve zbrani“ či „munice hráče“. UI je k vidění v pravém dolním rohu obrázku č. 16.

3.4 Tvorba audio efektu

O zvucích ve filmech a hrách je známo, že žádné z nich v realitě nezní tak jak jsou nám v nich prezentovány. Každý zvuk nejdříve prochází úpravou nebo je úplně nahrazen za zvuk uším příjemnější. Zvuky o vysokých decibelech a výškách jsou obtížnější na nahrání, a proto jsou zvuky jako třeba střelba zbraně nahrány na špičková zařízení, nebo jsou zaměněny za zvuky podobné.

K nahrání a úpravě audio do hry použijeme program Audacity. Nejprve se rozhodneme jaký zvuk budeme nahrávat. V tomto příkladu budeme nahrávat a upravovat zvuk střelby z pistole. Jak by se mohlo zdát, nahrání tohoto zvuku přísluší jen člověku se zbrojním průkazem na střelnici. To je ovšem omyl! Jelikož nahrání reálného výstřelu by bylo obtížnější, nahradíme ho zvukem podobným, a to zabouchnutím obyčejných dveří. Vstup mikrofону lehce ztlumíme a mikrofon přistavíme ke dveřím, spustíme nahrávání a hrubě bouchneme dveřmi. Toto opakujeme alespoň 3x, protože budeme mít větší možnost výběru při volbě toho správného záznamu. Záznam překontrolujeme a upravíme výšky audia. Na konec záznamu umístíme lehkou ozvěnu a postupně ztlumení zvuku. Výsledkem bude efekt, který budeme přehrávat při každém výstřelu ze zbraně.

3.5 Tvorba herního prostředí

Při tvoření herního prostředí je důležité si uvědomit, jaký příběh budeme chtít vyprávět. Pro svůj projekt jsem vytvořil nejdříve hrubý koncept scén, teprve pak jsem začal tvořit herní prostředí. Původní náčrt herního plánu je na obrázku č. 17

obrázek č.17

Když byl vytvořen koncept, můžeme začít umisťovat do herního prostředí tvz. „placeholder“, tedy dočasné objekty, které budou později nahrazeny za vymodelované objekty s texturami a kolizí. Díky tomuto kroku získáme představu o tom, jaké měřítko naše herní pole vlastně má a jak se v něm hráč bude cítit. Mapu s „placeholder“ můžeme vidět na obrázku č. 18.



obrázek č. 18

Dále nahrazujeme velké „placeholders“ jako jsou domy za plnohodnotné objekty. Budeme tedy v procesu nahrazování „placeholedrů“ postupovat od největších po nejmenší. Postupně přidáváme stále menší a menší detaily, které lépe dokreslují prostředí, v němž se nacházíme. Objekty nejsou vždy dobře umístěné, a tak je různě přesouváme dle potřeby. Když jsou všechny, nebo většina objektů na svém místě, můžeme se soustředit na náročnost renderování. Mým cílem bylo nenarazit na místo ve hře, kde by renderování objektů bylo natolik náročné, že by „fps“ (frames pre second) hry sahaly pod hodnotu 60fps. Toho jsem docílil optimalizací. Renderování malých objektů na velké vzdálenosti je zbytečné, tudíž jsem použil nástroj s názvem „Cull Distance Volume“, který zakazuje renderování objektů určité velikosti do nastavené vzdálenosti od objektu. Další možností optimalizace je použití LOD modelů, avšak z důvodu nedostatku času tato metoda nebyla v projektu využita. Konečný výsledek je zachycen na obrázku č. 19.



obrázek č. 19

4 POSTAVA HRÁČE

V konečném řešení je postava hráče schopna chůze pomocí kláves W,A,S,D. Sprint spouštěný pomocí klávesy shift není funkční. Levým kliknutím myši dojde k výstřelu z aktuálně vybrané zbraně, odečte se jeden náboj z munice nabité ve zbrani a aktuální počet nábojů je vykreslen na UI hráče. Zaměření skrz zaměřovač pravým tlačítkem myši není v provozu. Model rukou hráče má všechny animace pro pohyb, ovšem tyto animace nebyly zamýšleny jako konečné. Co se týče animací zbraní, nejsou implementovány do finálního řešení. Hráč je schopen hru pozastavit stisknutím kláves ESCAPE nebo P. Pause menu obsahuje tlačítka pro ukončení hry, návrat do hlavního menu, nebo návrat do hry. V seznamu příloh můžete naléznout obrázky s čísly 11-15, které jsou viditelnou součástí hlavní postavy hráče.

5 HERNÍ PROSTŘEDÍ

Herní prostředí mělo být původně plně interaktivní, mělo obsahovat zvuky města, dav lidí řízený umělou inteligencí, nepřátelskou umělou inteligenci, která by hráče ohrožovala různými způsoby. Dále jsem měl v plánu vložení krátkých animací při rozrážení dveří hráčem, komunikaci mezi hráčem a postavami, která by dokreslovala příběh. Do splněných cílů, co se týče herního prostředí, mohu zařadit předem vytyčenou cestu, po které hráč prochází „level“. Hráč se nemůže vychýlit mimo určenou trasu díky polím zamezujícím pohyb mimo trasu. Herní mapa má 7 různých typů budov, aby ovšem vypadalo prostředí přirozeně, bylo by jich zapotřebí alespoň 20. Na domech mohou být různé modifikace jako satelit, klimatizace, mříže na oknech, apod. Tyto detaily totiž napomohou dojmu, že každý dům je originální. Pro dokreslení prostředí jsem použil detaily jako pohozené odpadky, volně ležící dřevěná prkna či plechy. Velkou roli také hrají sloupy s elektrickým vedením a lampy pouličního osvětlení, bez nich jsou ulice příliš prázdné a nereálné. Co se týče automobilů, měl jsem v plánu vytvořit 5 různých typů, avšak kvůli nedostatku času je umístěn na mapě pouze jeden typ. Ve výsledku je v herním prostoru nad 4000 objektů, z toho přes 100 různých. Vše stojí na podloží, které má 4 různé vrstvy materiálů. Obrázky finálního provedení můžete vidět na obrázcích v seznamu příloh s čísly 1 – 10.

Závěr

Projekt byl poměrně náročný, a i přesto že jsem na něm začal pracovat o měsíce dříve než mí kolegové, nepodařilo se mi dosáhnout všech vytyčených cílů. Mezi ty splněné ovšem patří plně průchozí „level“ hry, ovladatelné hlavní menu, schopnost hráče se pohybovat, střílet a přebíjet zbraně. Co se týče nesplněných cílů a dalších vylepšení, pracoval bych dále na příběhu hry, animacích a umělé inteligenci nepřátelských jednotek. Hra se jednoznačně může dostat do fáze, kdy bude schopna vydání na platformách jako Game Jolt nebo Steam.

SEZNAM POUŽITÝCH INFORMAČNÍCH ZDROJŮ

- [1] *Dokumentace pro Unreal Engine 4* [online]. [cit. 2020-01-13]. Dostupné z:
<https://docs.unrealengine.com/en-US/Programming/index.html>
- [2] *Wikipedie pro Unreal Engine 4* [online]. [cit. 2020-01-13]. Dostupné z:
<https://wiki.unrealengine.com>
- [3] *C++ reference* [online]. [cit. 2020-01-13]. Dostupné z:
<http://www.cplusplus.com/>
- [4] *Dokumentace pro Visual Studio 2017* [online]. [cit. 2020-01-13]. Dostupné z:
<https://docs.microsoft.com/en-us/visualstudio/?view=vs-2019>
- [5] *Logs, Printing Messages To Yourself During Runtime* [online]. [cit. 2020-01-13].
Dostupné z:
https://wiki.unrealengine.com/Logs,_Printing_Messages_To_Yourself_During_Runtime
- [6] *Collision Filtering* [online]. [cit. 2020-01-13]. Dostupné z:
<https://www.unrealengine.com/en-US/blog/collision-filtering>
- [7] *Rendering and Graphics* [online]. [cit. 2020-01-13]. Dostupné z:
<https://docs.unrealengine.com/en-US/Engine/Rendering/index.html>
- [8] *GitHub* [online]. [cit. 2020-01-13]. Dostupné z:
www.github.com/CubeCodes/TestingGrounds/invitations

SEZNAM PŘÍLOH

Obrázek č. 1 Prostředí hry 1

Obrázek č. 2 Prostředí hry 2

Obrázek č. 3 Prostředí hry 3

Obrázek č. 4 Prostředí hry 4

Obrázek č. 5 Prostředí hry 5

Obrázek č. 6 Prostředí hry 6

Obrázek č. 7 Prostředí hry 7

Obrázek č. 8 Prostředí hry 8

Obrázek č. 9 Prostředí hry 9

Obrázek č. 10 Prostředí hry 10

Obrázek č. 11 Zbraň M4

Obrázek č. 12 Holografický zaměřovač

Obrázek č. 13 Zásobník zbraně M4

Obrázek č. 14 Hodinky Houdini

Obrázek č. 15 Model ruky hráče

Obrázek č. 1 Prostředí hry 1



Obrázek č. 2 Prostředí hry 2



Obrázek č. 3 Prostředí hry 3



Obrázek č. 4 Prostředí hry 4



Obrázek č. 5 Prostředí hry 5



Obrázek č. 6 Prostředí hry 6



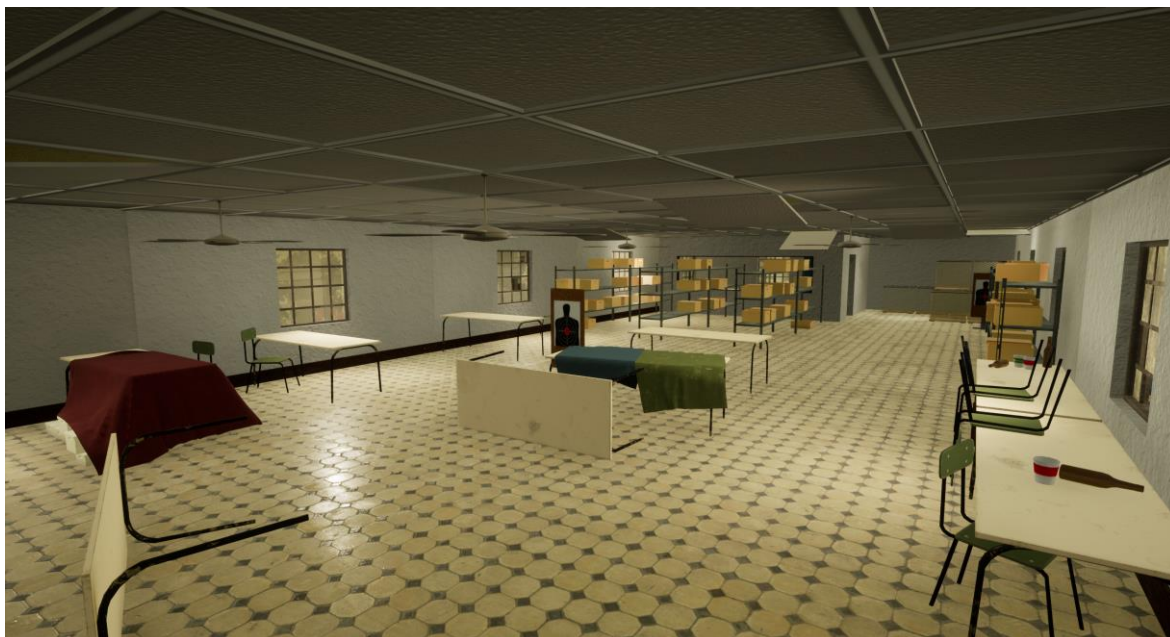
Obrázek č. 7 Prostředí hry 7



Obrázek č. 8 Prostředí hry 8



Obrázek č. 9 Prostředí hry 9



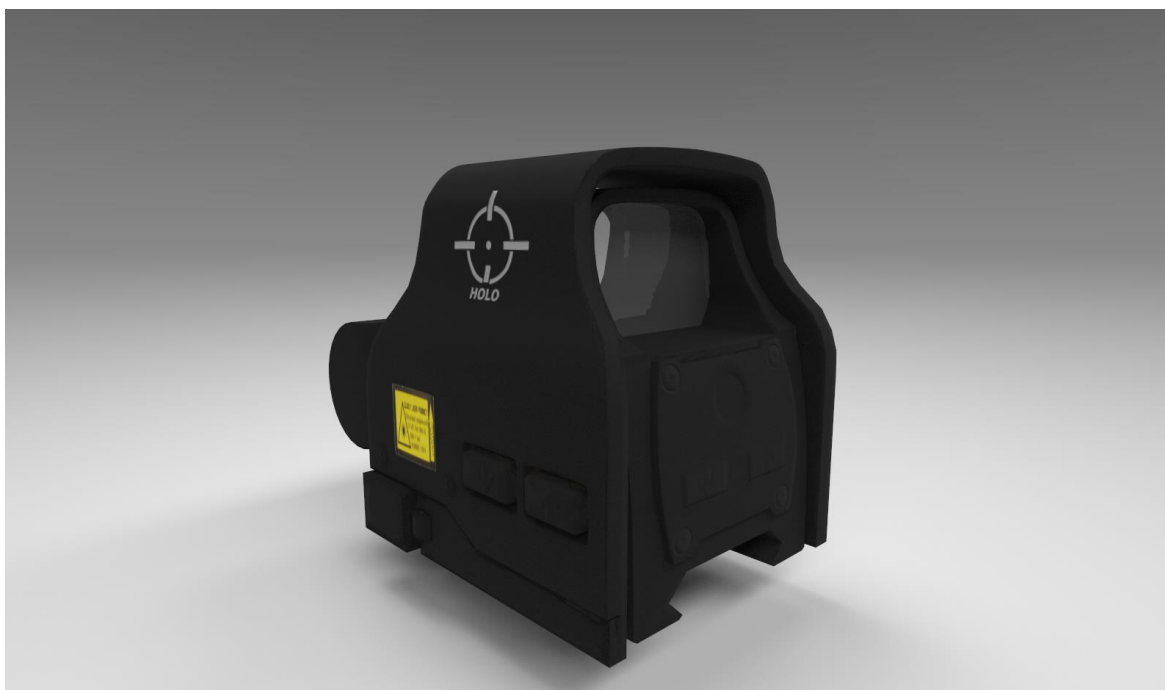
Obrázek č. 10 Prostředí hry 10



Obrázek č. 11 Zbraň M4



Obrázek č. 12 Holografický zaměřovač



Obrázek č. 13 Zásobník zbraně M4



Obrázek č. 14 Hodinky Houdiny



Obrázek č. 15 Model ruky hráče

