

Vývoj aplikace v 3D engine

Aladár Knís

Bakalářská práce
2019



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2018/2019

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Aladár Knís**
Osobní číslo: **A18477**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**
Forma studia: **kombinovaná**

Téma práce: **Vývoj aplikace v 3D engineu**
Téma anglicky: **Application Development in 3D Engine**

Zásady pro vypracování:

1. Všeobecně analyzujte vývojové prostředí 3D engineů, jejich základní vlastnosti, technologie a možnosti implementace na různé platformy.
2. Porovnejte nejčastěji používané 3D enginey od různých vývojářů z hlediska použitých technologií a nástrojů.
3. Analyzujte možné využití aplikací vytvořených v 3D enginech v různých oblastech.
4. Na základě získaných poznatků, pro vybraný engine navrhnete vlastní aplikaci, která bude využívat většinu technologií tohoto engineu a zároveň je bude demonstrovat.
5. Příslušnou aplikaci v tomto engineu naprogramujte a odladte.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **COOKSON, Aram, Ryan DOWLINGSOKA a Clinton CRUMPLER. Sam's teach yourself Unreal engine 4 game development in 24 hours. Indianapolis, Indiana: Sams, [2016]. Sams teach yourself in 24 hours series. ISBN 0672337622.**
2. **Unreal Engine 4 Documentation [online]. [cit. 2018-11-28]. Dostupné z: <https://docs.unrealengine.com/en-us/>**
3. **LEE, Joanna. Learning Unreal Engine Game Development. 2016. ISBN 9781784398156.**
4. **SEWELL, Brenden. Blueprints Visual Scripting for Unreal Engine. 2015. ISBN 9781785286018.**
5. **PV, Satheesh. Unreal Engine 4 Game Development Essentials. 2016. ISBN 9781784391966.**
6. **GREGORY, Jason. Game engine architecture. Second edition. Boca Raton: CRC Press, 2014. ISBN 978-1466560017.**

Vedoucí bakalářské práce:

Ing. Pavel Pokorný, Ph.D.

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

21. prosince 2018

Termín odevzdání bakalářské práce:

15. května 2019

Ve Zlíně dne 21. prosince 2018

doc. Mgr. Milan Adámek, Ph.D.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne

.....
podpis diplomanta

ABSTRAKT

Cílem této práce bude obecně analyzovat vývojové prostředí 3D engine, jeho základní vlastnosti, moderní technologie a implementace na různých platformách. Dále student porovná v současnosti nejčastěji používané 3D enginey od různých vývojářských týmů z hlediska implementovaných nástrojů a podporovaných technologií. Na základě získaných poznatků v praktické části s pomocí vybraného engine navrhne a vyvine vlastní aplikaci, která bude využívat nejmodernější technologie používané v komerčním průmyslu. Vizuální výstup této aplikace navrhne tak, aby uvedené technologie názorně demonstrovala.

Klíčová slova: Unreal Engine 4, Herný Engine, Asset, Blender

ABSTRACT

The purpose of this work is to analyze integrated development environment of 3D engines in general, implemented technologies and possible platforms. Then comparison of the most used 3D engines from different developers by the implemented set of tools and supported technologies.

Based on these information, practical part focuses on design and development of the application in the chosen engine with the use of the most modern technologies used in commercial business. The visual output of this application is designed to demonstrate the stated technologies.

Keywords: Unreal Engine 4, Game Engine,

POĎAKOVANIE

Chcem sa poďakovať pánovi Ing. Pavlovi Pokornému, Ph.D. za jeho trpezlivosť, rady, tipy a pripomienky počas tvorby mojej práce a za jeho pohotovú komunikáciu.

Ďalej sa chcem poďakovať svojim rodičom a blízkym za ich podporu a dôveru počas môjho štúdia.

PREHLÁSENIE

Čestne prehlasujem, že odovzdaná verzia bakalárskej práce a elektronická verzia nahraná do IS/STAG sú totožné

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 ČO JE 3D ENGINE	11
1.1 HISTÓRIA A VÝVOJ	11
1.2 POPIS PROSTREDIA	12
1.3 TECHNOLOGIE	13
1.3.1 Grafický rendering	13
1.3.2 Editor 3D objektov	14
1.3.3 Simulácia fyziky.....	14
1.3.4 Sieťové nástroje.....	14
1.3.5 UI design	14
1.3.6 Logika a umelá inteligencia	14
1.4 PLATFORMY	14
1.4.1 Počítače	14
1.4.2 Herné konzoly	15
1.4.3 Smartfóny	15
1.4.4 Multi-platformy.....	15
1.5 KOMPONENTY HERNÝCH ENGINOV	16
1.5.1 Grafický engine	16
1.5.2 Zvukový engine.....	16
1.5.3 Fyzikálny engine	17
1.5.4 Input manažér	17
1.5.5 Sieťová infraštruktúra	17
1.6 POPIS INTERAKCIE MEDZI JEDNOTLIVÝMI KOMPONENTMI	17
1.7 PRINCÍP INTERAKCIE MEDZI JEDNOTLIVÝMI KOMPONENTMI	19
1.8 ASSETY HERNÉHO ENGINU	19
1.8.1 Mapy	20
1.8.2 Textúry a materiály	20
1.8.3 Zvuky	20
1.8.4 Statické meshe.....	21
1.8.5 Animácie	21
2 POROVNANIE NAJPOUŽÍVANEJŠÍCH 3D ENGINOV	22
2.1 UNREAL ENGINE 4.....	22
2.2 UNITY 3D.....	22
2.3 CRYENGINE 5.....	23
2.4 GODOT	23
3 POUŽITE	25
3.1 KOMERČNÝ PRIEMYSEL.....	25
3.2 TECHNICKÝ PRIEMYSEL.....	26
3.3 EDUKÁCIA	26
3.4 ARMÁDA	29
II PRAKTICKÁ ČÁST	31

4	VÝBER VHODNÉHO PROJEKTU PRE DEMONŠTRÁCIU	32
5	VÝBER VHODNÉHO 3D ENGINU	33
5.1	PROGRAMOVACÍ JAZYK	33
5.2	PROSTREDIE	33
5.3	OSOBNÝ DOJEM	33
5.4	VÝSLEDOK	34
6	POSTUP	35
6.1	IMPORT SCÉNY DO ENGINU	35
6.2	ÚPRAVA SCÉNY V HERNOM ENGINE	37
6.3	KAMERA A POSTAVA HRÁČA	38
	MATERIÁLY	40
6.4	OPTIMALIZÁCIA	40
6.5	VÝSLEDOK A ZHODNOTENIE	40
7	MOŽNOSTI PRE DALŠÍ ROZVOJ	41
7.1	VYTVORENIE SCÉNY V HERNOM ENGINE	41
7.2	VIRTUÁLNA A ROZŠÍRENÁ REALITA	41
7.3	OPTIMALIZÁCIA	41
7.4	NÁPLŇ	41
7.5	UŽÍVATELSKÉ ROZHRAŇIE	41
	ZÁVER	42
	ZOZNAM POUŽITEJ LITERATÚRY	43
	ZOZNAM POUŽITÝCH SYMBOLOV A SKRATIEK	44
	ZOZNAM OBRÁZKOV	45
	ZOZNAM TABULIEK	46
	ZOZNAM PRÍLOH	47

ÚVOD

Herný engine je softwarový nástroj, poskytujúci základnú funkcionálnosť, potrebnú na tvorbu videohry. V posledných rokoch, a hlavne s príchodom virtuálnej a rozšírenej reality, sa však ale od tejto definície upúšťa a hlavne od toho, že jeho výstup môže byť jedine videohra.

Hoci videohry zväčša slúžia len pre zábavu a odreagovanie a často sú vnímané ako plytvanie času a zdrojov, v oblasti informačných technológií spôsobili veľký prelom a boli a sú hnačím motorom počítačovej 3D grafiky, či už zo strany hardwaru alebo softwaru.

Dnes sa tieto technológie využívajú v armáde, zdravotníctve, školstve, marketingu a priemysle a stále sa prichádza na nové možnosti ich využitia.

Väčšina herných enginev dokáže renderovať 2D aj 3D tvary, importované zo softwaru tretej strany, ako napríklad Blender, 3D Studio Max alebo Maya. Ďalšou významnou funkciou je možnosť programovania kódu, ktorý potom herný engine vykonáva. Vďaka tomu herný engine získava presah nad grafickými editormi, keďže 3D modely dostávajú funkcionálnosť a logiku.

V mojej práci by som sa chcel zamerať na porovnanie dostupných herných enginev na základe technológií a programovacích jazykov, ktoré podporujú, popísal moje skúsenosti s nimi a dojem, ktoré na mňa zanechali ako na používateľa. Ďalej by som rád preskúmal a popísal možnosti využitia aplikácie vytvorenej v hernom engine v rôznych odvetviach a odboroch. V praktickej časti svojej práce sa zameriam na výber vhodných technológií a projektu pre demonštráciu možností herného engineu.

I. TEORETICKÁ ČASŤ

1 ČO JE 3D ENGINE

1.1 História a vývoj

Herné enginy majú relatívne krátku históriu v porovnaní s informatikou vo všeobecnosti, zato ich vývoj bol rýchly a dynamický.

Počínajúc v 90. rokoch, kedy sa procesor 486 stal dostupným pre bežných ľudí a jednalo sa o procesor dostatočne rýchly, aby sprístupnil 3D hry pre všetkých. Nejednalo sa ale úplne o pravé 3D, ale o pseudo-3D, čo sú vlastne 2D obrázky vykreslené v tvare trojdimenzionálnej geometrie. Technológie pre pravé 3D boli pre bežného o užívateľa vzdialené ešte niekoľko rokov.

Pseudo-3D využívalo škálovanie a skresľovanie 2D obrázkov, aby vzbudili dojem, že sú v 3D priestore, ale pravá vertex-transformácia sa však neobjavila v populárnych herných enginách až do roku 1995, s príchodom hry Quake 1, a to z veľkého ohľadu aj vďaka rýchlejšim a vylepšeným grafickým procesorom.



Obrázok 1 Pseudo-3D hra

Predstavenie výkonných grafických výpočtových jednotiek (GPU) otvorilo možnosti novým 3D hráčom. V počiatku fungovali len na počítačoch v hodnote niekoľkých miliónoch USD, ale vďaka 3dfx, Nvidia a Maxtronx, v neskorých 90. rokoch, sa GPU stali nutnosťou pre väčšinu 3D hier. V tomto čase, spoločnosť id Software, vedená Johnom Carmackom, sa dostala rýchlo do popredia predstavením technológií ako spline surfaces (pokrivené plochy), komplexné tieňovanie a osvetlenie vo svojej novej hre Quake 2. Okolo roku 1998 predstúpili do popredia Epic Games so svojím Unreal Engine a Valve Software so Source Engine, vydaním hier Unreal Tournament a Half-Life.

Pred tým, ako boli dostupné dedikované GPU a lacné RAM, bežné počítače nemali dostatok zdrojov pre podporu 3D prostredia a často hry využívali počítač natoľko, že ostávalo len veľmi málo miesta pre niečo navyše, ako napríklad časticové efekty a just-in-time kompiláciu.

Tieto technológie boli z veľkej časti ovplyvnené „modovacíou“ [mod od slova modifikácia – pozn. autora] komunitou [bežní užívatelia, vytvárajúci vlastný obsah za pomoci dostupných technológií – pozn. autora], keďže donútili vývojárov, aby ich technológie boli ústretivejšie k ich komunite a pridali viacero user-friendly nástrojov.

Nie je prekvapením, že viacero vývojárov sa snažilo podchytiť si túto komunitu a taktiež aj nezávislých vývojárov a niektoré spoločnosti ako RenderWare a Gamebryo, ťažko zamerané hlavne na veľkých vydavateľov, už nedokázali konkurovať nováčikom ako Unity 3D.

V duchu nezávislého vývoja sa mnoho herných enginov zameralo na dostupný a multi-platformový vývoj a nástroje ako Game Maker, UDK, Unity 3D a Unreal Engine sprístupnili ľuďom možnosti učiť sa a vyvíjať hry bez miliónových investícií. [1]

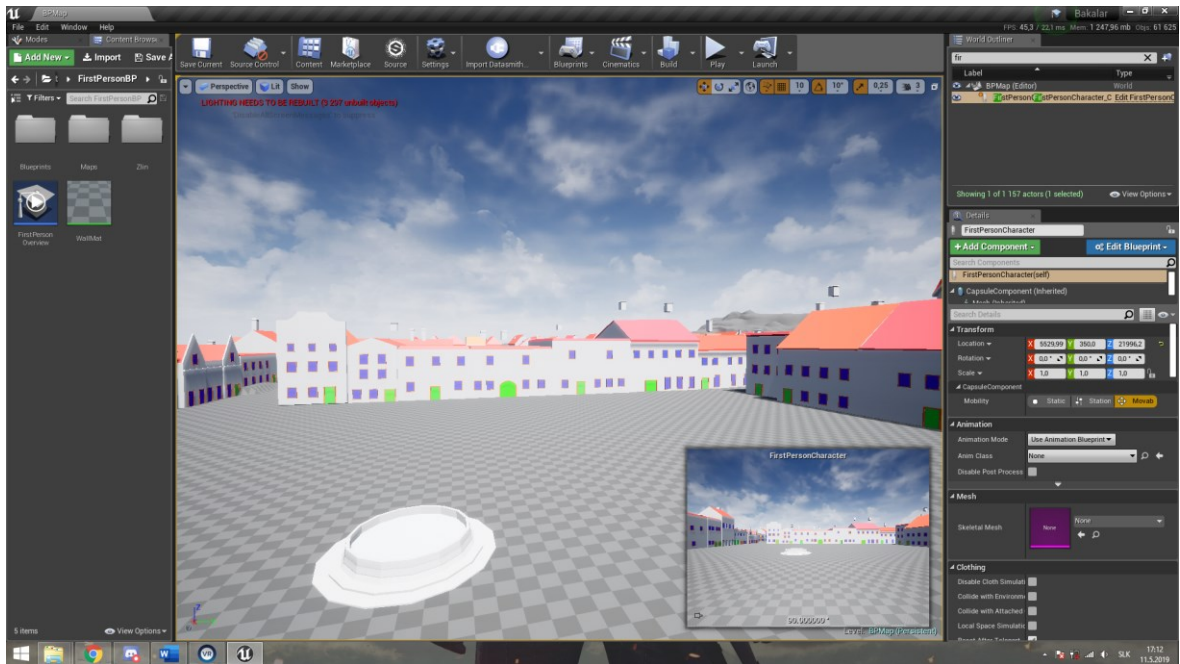
1.2 Popis prostredia

Pri popise budeme vychádzať z predvolených nastavení, keďže prostredie umožňuje vlastné usporiadanie.

Centrálnym prvkom každého herného enginu je hlavné okno, poskytujúce náhľad do vyvíjanej aplikácie. V tomto okne je možné upravovať, odoberať, pridávať a transformovať jednotlivé objekty. Taktiež je možné označiť konkrétny objekt a ďalej pokračovať v detailnejšej úprave. Veľmi podstatnou vlastnosťou tohto okna je testovací režim, ktorý umožňuje spustenie a potom prípadné zapauzovanie vyvíjanej aplikácie bez nutnosti kompilácie celého projektu.

Bočnými alebo okrajovými prvkami sú prehliadače obsahu, ktorý je možné vložiť do aplikácie, hierarchický model aplikácie a editor vlastností vybraného objektu.

Prostredie využíva systém záložiek, podobne ako internetový prehliadač. Prvky vyžadujúce celú obrazovku sa otvárajú ako nová záložka. Spravidla sa jedná o nástroje pre úpravu 3D modelov, alebo pri Unreal Engine 4 sa jedná o editor programov v grafickom programovacom jazyku blueprint. [9]



Obrázok 2 Ukážka prostredia Unreal Engine 4

1.3 Technológie

3D engine poskytuje vývojárovi kompletnú sadu nástrojov, ktoré sú späté s rôznymi technológiami, a to pod jednou strechou. Zahŕňa v sebe technológie 3D grafických editorov, vývojových prostredí IDE, animačných nástrojov a audio editorov. Tieto technológie môžeme rozdeliť do niekoľkých bodov: [2]

1.3.1 Grafický rendering

Základnou technológiou je grafický rendering, a to hlavne v reálnom čase a s tým spojené optimalizačné nástroje. [2]

1.3.2 Editor 3D objektov

Editor 3D objektov poskytuje takmer plnohodnotnú funkcionálnu grafických editorov, vrátane úpravy jednotlivých polygónov a UV máp. Taktiež má kompletnú sadu animačných nástrojov. [2]

1.3.3 Simulácia fyziky

Každému 3D objektu je možné priradiť a upravovať jeho fyzikálny model. [2]

1.3.4 Sieťové nástroje

3D engine má v sebe predimplementované nástroje pre sieťovú komunikáciu a streamovanie dát, čím výrazne uľahčuje vývoj aplikácií typu klient. [2]

1.3.5 UI design

Opäť sa jedná o funkcionálnu grafických alebo konkrétne UI editorov. Služi zväčša na prototypovanie prvkov užívateľského rozhrania. [2]

1.3.6 Logika a umelá inteligencia

Okrem rozhodovacích stromov a sekvencií, majú v sebe 3D engine implementované aj základné prvky umelej inteligencie, ako napríklad hľadanie ciest. [2]

1.4 Platformy

3D engine dávajú vývojárom pestrý výber cieľovej platformy. Vďaka tomu, že dnes máme vo vrecku zariadenie s výkonom počítača z minulého desaťročia, otvorili sa možnosti vývoja graficky náročných aplikácií pre mobilné zariadenie, čo prináša potenciál, ktorý nie je ešte do dnešného dňa preskúmaný.

1.4.1 Počítače

Jedná sa o platformu ktorá slúži aj ako vývojová, ale aj cieľová platforma.

Podporované sú všetky hlavné operačné systémy, menovite: Windows, Linux a Mac OS. Nutnou podmienkou je samozrejme splnenie minimálnych požiadaviek, čo je aktuálnosť operačného systému a dostatočný hardware. Pokiaľ ide o vývoj virtuálnej alebo rozšírenej reality, môžu byť tieto požiadavky vyššie, keďže sa jedná o náročnejšiu aplikáciu. Pri vývoji

je důležité sa rozhodnúť, pokiaľ bude siahať spätná kompatibilita a pre aké konfigurácie bude aplikácia optimalizovaná. [2]

1.4.2 Herné konzoly

Keďže herné konzoly sú kompaktné zariadenia a majú jasne daný hardware a odpadajú pri nich hardwarové požiadavky, aplikácia musí byť ale perfektne optimalizovaná na ich hardware, a to hlavne pri komplexnejších projektoch a virtuálnej realite.

Podporované sú samozrejme všetky hlavné konzoly, menovite: Playstation 4, Xbox One, Nintendo Switch. [2]

1.4.3 Smartfóny

Pri vývoji pre smartfóny veľmi záleží na operačnom systéme telefónu. Pri Androide je vývoj podobný ako pri vývoji na PC, keďže existuje kvantum rôznych konfigurácií hardwaru a softwaru, takže je nutné si určiť rozumnú minimálnu hranicu kompatibility.

Pri vývoji pre iOS sa vývoj podobá viac vývoju na konzolu, keďže sa jedná o jasne daný hardware a software, treba si ale určiť minimálny podporovaný model, keďže vychádzajú periodicky každé 2 roky.

Pokiaľ sa jedná o vývoj aplikácií s virtuálnou alebo rozšírenou realitou, jedná sa zväčša o najnovšie modely, keďže už samotná technológia má svoje minimálne požiadavky, či už na verziu operačného systému, hardwaru alebo potrebných senzorov, hlavne je nutná prítomnosť gyroskopu. [2]

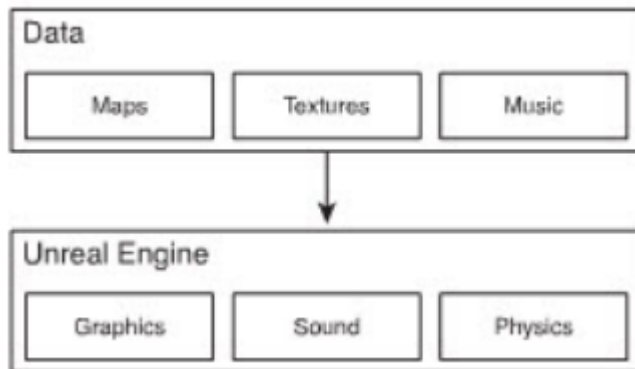
1.4.4 Multi-platformy

Najrozšírenejšou technológiou pre multiplatformový vývoj je HTML5 a WebGL. Pre spustenie aplikácie, využívajúcej túto technológiu, je potrebný len internetový prehliadač. Samozrejme je nutné použiť prehliadač, ktorý tieto technológie podporuje a mať ho príslušne aktualizovaný.

Pri vývoji na túto platformu je optimalizácia najväčšou prioritou. Pravdepodobne nie je výhodná pre náročné aplikácie, keďže nie je možné odhadnúť, na akých zariadeniach bude spúšťaná, zato ale má jasné výhody v dostupnosti. [2]

1.5 Komponenty herných enginev

Engine je rozdelený do troch nezávislých častí spojených hlavným jadrom. Jednotlivé časti sú modulovateľné a nahraditeľné vlastnými, bez nutnosti zmeny hlavného jadra. [3]



Obrázok 3 Komponenty herného enginev

1.5.1 Grafický engine

Grafický engine riadi to, čo v aplikácii vidíme a kalkuluje, ktoré objekty sa zobrazia v popredí a ktoré v pozadí. Objekty, ktoré sú v pozadí za inými objektmi, sú označované statusom okludované a grafický engine ich v tomto prípade vôbec nerenderuje, čím aplikáciu od základu optimalizuje. Engine obsahuje technológiu level streaming, ktorá umožňuje načítavať a uvoľňovať z pamäti jednotlivé sublevely počas behu aplikácie, takže, ak užívateľ opustí jeden sublevel a vstúpi do ďalšieho, predošlý sa uvoľní z pamäti a nebude viac renderovaný grafickým enginev. Vo výsledku to znamená, že užívateľ môže prechádzať scénou s veľkou hustotou objektov bez nutnosti načítavania.

Grafický engine taktiež renderuje tieň, materiály, osvetlenie pre rôzne objekty na scéne. [3]

1.5.2 Zvukový engine

Zvukový engine zoraďuje všetky importované zvukové súbory na základe eventov v aplikácii, ktoré potom čo najpresnejšie rekonštruuje na základe poskytnutého hardwaru PC alebo konzoly. Celý systém funguje na takzvaných SoundCues, čo je list inštrukcií, ktorý kontroluje, ako budú jednotlivé zvukové súbory prehrané a poskytuje funkcie ako slučka, zmena tónu, modulácia zvuku a kombinovanie viacerých zvukov do seba. [3]

1.5.3 Fyzikálny engine

Herný engine využíva rôzne technológie, ale najčastejšie technológiu PhysX od spoločnosti Nvidia, ktorá poskytuje kvantum fyzikálnych simulácií. Najvýznamnejšie sú dynamické skeletové meshe. Každá z týchto interakcií môže byť manipulovaná počas behu aplikácie, a to priamo alebo cez naskriptované sekvencie.

Fyzikálny engine spolupracuje so zvukovým enginom, aby generoval dynamické zvuky, keď dôjde ku kolízii dynamických objektov. [3]

1.5.4 Input manažér

Vždy, keď užívateľ odošle vstup, input manažér nastaví sériu udalostí, ktoré sa majú odohrať v aplikácii. Pri každom stlačení nastavenej klávesy input manažér pošle signál do jadra enginu, čo spôsobí priradenú animáciu, grafický engine vykreslí danú akciu na obrazovku a prehrá priradený zvukový efekt zo zvukového enginu. Všetky tieto operácie nastanú vo veľmi krátkom čase a užívateľ nepostrehne žiadne oneskorenie medzi inputom a reakciou aplikácie. Input manažér je veľmi flexibilný a vďaka tomu uľahčuje zmenu platformy. [3]

1.5.5 Sieťová infraštruktúra

Mnoho enginov pôvodne vzniklo ako herný engine pre hru viacerých hráčov, takže sieťová komunikácia je vysoko optimalizovaná.

Aplikácie pre viacerých užívateľov môžu byť založené na dvoch modeloch komunikácie.

Jeden je, že jeden užívateľ sa pripojí na druhého užívateľa, takže druhý užívateľ je zároveň aj klientom, aj serverom. Druhý model je, že užívatelia sa pripájajú na systém, ktorý je striktno server a neobsahuje klienta, tento model sa nazýva dedikovaný server.

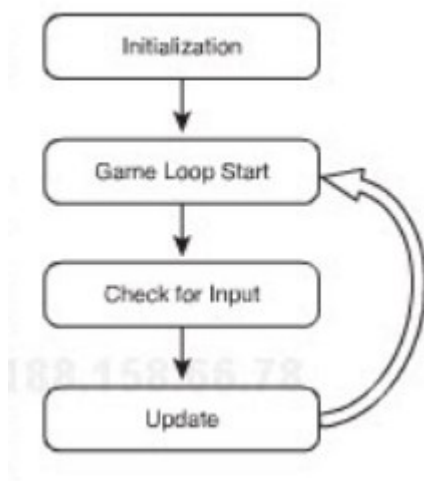
Kľúčom k efektívnej sieťovej komunikácii je v limitovaní dát, posielaných cez sieť na len tie najnutnejšie aspekty aplikácie, ako pozície a interakcie s prostredím. [3]

1.6 Popis interakcie medzi jednotlivými komponentmi

Väčšina aplikácií v hernom engine funguje na základe takzvanej hernej slučky, časti kódu aplikácie, ktorá sa konštantne opakuje, sleduje input používateľa a obnovuje všetky aspekty aplikácie. Vo vnútri slučky engine opakovane vykonáva niekoľko individuálnych kontrol toho, čo sa zmenilo v prostredí aplikácie.

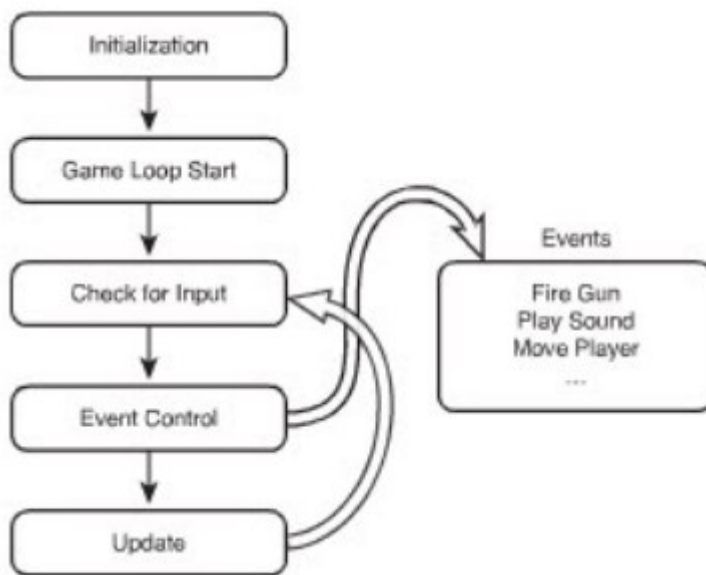
Herná slučka zaraďuje úlohy jednotlivým súčastiam enginu, ako napríklad prekresľovanie obrazovky grafickým enginom, prehrávanie zvukov zvukovým enginom a odosielanie dát cez sieť.

Herné slučky sú dizajnové lineárne, čo znamená, že každý cyklus prechádza cez sériu v presne rovnakom poradí, zakaždým s rovnakou prioritou. Niektoré enginy majú ale rozvi- nutejšiu funkcionality hernej slučky.



Obrázok 4 Herná slučka

Napríklad Unreal Engine používa hernú slučku založenú na eventoch, čo znamená, že engine obsahuje list udalostí, ktoré rôzne komponenty musia vykonať. Tieto udalosti môžu pochádzať od rôznych zdrojov, ako napríklad input užívateľa, dáta z fyzikálneho enginu alebo komunikácia medzi komponentmi. Všetko je predávané do systému cez list udalostí.



Obrázok 5 Herná slučka založená na udalostiach

Pri tejto metóde je každej udalosti priradená špecifická priorita a namiesto obnovovania všetkých aspektov aplikácie počas každej otočky cyklu, udalosti sú vyhodnocované na základe dôležitosti. [3]

1.7 Princíp interakcie medzi jednotlivými komponentmi

Po spustení enginu sa všetky komponenty inicializujú. Počas inicializácii začnú medzi sebou jednotlivé komponenty komunikovať a jadro enginu začne rozosielať príkazy pre jednotlivé komponenty. Tie potom odosielajú dáta naspäť jadrú a udržujú synchronizáciu.

Po tom, ako sa všetky komponenty inicializujú a používateľské rozhranie je zobrazené, Engine čaká na input.

Po spustení aplikácie jadro načíta level, ktorý obsahuje všetky assety nutné pre chod aplikácie, čo sú: textúry, materiály, zvuky, meshe, animácie a skripty. Každý asset je spárovaný so zodpovedajúcim komponentom. Materiály sú spárované s grafickým enginom, zatiaľ čo dynamické meshe sú spárované s fyzikálnym enginom.

V tomto bode sa inicializuje poradovník udalostí a jadro ho naplní dátami. [3]

1.8 Assety herného enginu

Virtuálne všetky dáta v hernom engine sú zabalené do balíčkov, čo sú vlastne kolekcie assetov, ktoré sú potrebné pre projekt. Balíčky sú rozdelené podľa typov assetov, ktoré obsahujú. [3]

V nasledujúcich podkapitolách budú jednotlivé typy popísané.

1.8.1 Mapy

Mapy, taktiež nazývané aj levely alebo scény, je miesto, kde je vytvorené prostredie aplikácie v engine.

Technicky je mapa kolekciou rôznych assetov, ktoré sú zobrazované používateľovi, kdežto väčšina týchto assetov je načítavaná z vytvorených balíčkov. Esteticky je mapa scénou pre vizuálny zážitok používateľa. Je to miesto, kde sa vytvára celkový vzhľad prostredia, dodáva sa celému projektu život.

Každá mapa obsahuje svoj vlastný balíček, kam sú ukladané všetky assety, ktoré sú použité len v konkrétnom leveli. Assety, ktoré sú uložené v balíčku mapy a neboli využité, sú pri uložení mapy z tohto balíčku odstránené. [3]

1.8.2 Textúry a materiály

Textúry a materiály sú ukladané v rovnakom balíčku.

Textúry sú 2D obrázky používané na tvorbu materiálov v leveloch a ostatných assetoch.

Každá viditeľná plocha má svoj materiál a každý materiál má aspoň jednu textúru, takže textúra je súčasťou materiálu, obsahujúca farbu a ostatné dôležité informácie.

Materiál teda obsahuje textúry a inštrukcie pre vykresľovanie.

Niektoré textúry, napríklad normálové mapy, prezentujú ilúziu, že povrch má viac fyzických detailov, než dokáže vykresliť daný hardware. Vďaka tomu modely, ktoré obsahujú len niekoľko tisíc polygónov, sa môžu javiť, ako keby ich mali mnohonásobne viac.

Tvorba textúr nie je súčasťou herného engine, keďže na to slúžia 2D grafické programy, napríklad Photoshop. Taktiež je možné použiť obrázky zachytené digitálnym fotoaparátom, alebo skenerom. [3]

1.8.3 Zvuky

Rovnako ako textúry, zvuky môžu byť vytvorené len mimo herný engine.

Zvuky sú spravidla prehrávané na podnet nejakej udalosti v aplikácii, ale niektoré zvuky môžu byť prehrávané počas celého behu aplikácie, takéto zvuky sa nazývajú ambientné.

Po importe zvukových súborov, SoundCues poskytujú veľkú flexibilitu využitia. Je možné mixovať niekoľko zvukov do jedného, náhodné prehrávanie zvukov, zmena hlasitosti, rýchlosti, modulácia a individuálne efekty.

Použitím skriptov je možné vytvoriť komplexné zvukové systémy, ktoré prehrávajú jednotlivé zvuky na základe špecifických udalostí. [3]

1.8.4 Statické meshe

V 3D grafike sú objekty tvorené z polygónových meshov – kolekcií verticlov, rohov a tvárí, ktoré definujú tvar objektu. Rôzne typy polygónových meshov sa správajú rozlične.

Statické meshe majú výhodu rýchleho vykresľovania a je možné z nich vytvoriť komplexné tvary, čo znamená, že sú vhodné pre herné enginy. Nedokážu byť ale použité na všetko.

Statické meshe sú dizajnované na niekoľkonásobné inšancovanie a reprodukovanie vo vnútri grafickej karty a nevyžadujú žiadny výkon procesoru, čo znamená, že sú vykresľované extrémne rýchlo, aj pri veľkom počte, spolu s faktorom toho, že je možné niektoré využiť znovu, namiesto vytvárania nových.

Veľkú časť pamäte využíva len prvá inštancia, kdežto pri ostatných grafická karta uchováva len informáciu o ich polohe, rotácii a škále a vykresľované informácie nie sú znovu kopirované. [3]

1.8.5 Animácie

Nie všetky objekty v aplikácii sú statické meshe. Sú to takzvané skeletové meshe, ktoré sa deformujú na základe digitálneho skeletu. Proces vytvárania a priradovania skeletu sa nazýva rigging. Taktiež je to proces, ktorý prebieha mimo herného enginu, typicky v 3D aplikácii, ako je Blender alebo Maya. Akonáhle je tento proces dokončený, môže byť skeletový mesh animovaný podľa potreby. [3]

Meshe a ich korešpondujúce animačné slučky sú uložené v balíčkoch. Animácie sú uložené zvlášť ako odlišný typ assetu, ktorý môže byť priradený akémukoľvek objektu s podobnou skeletovou štruktúrou. Tento systém uľahčuje animáciu, je šetrnejší k zdrojom, keďže redukuje počet animácii, ktoré musia byť načítané do pamäti. [3]

2 POROVNANIE NAJPOUŽÍVANEJŠÍCH 3D ENGINOV

2.1 Unreal Engine 4

Unreal Engine je vyvíjaný spoločnosťou Epic Games. Momentálne je jednotkou na trhu v realistickej vizualizácii a porastu a tvorbe terénu a je vhodný pre väčšie projekty.

Zato ale nie je úplne vhodný na vývoj malých mobilných hier a aplikácií, hoci podporuje platformy ako iOS a Android. Unreal obsahuje systém blueprint pre vizuálne skriptovanie, z prepojených blokov tvoriacich graf.

Unreal Engine je opensource a je vytvorený v C++, ktorý je aj jeho programovacím jazykom. Touto kombináciou poskytuje vývojárom kompletnú kontrolu nad celým systémom.

Jeho framework sa môže zdať komplikovaný a zložitý na pochopenie a učenie.

Najväčším benefitom Unreal Engine sú jeho vykresľovacie vlastnosti, výkone post-process efekty a široká podpora funkcií. Obsahuje taktiež editor materiálov, nástroje pre optimalizáciu a vizuálne odlaďovanie programu.

Jeho nevýhodou je, že jeho vývojári nie sú rovnako aktívny na komunitných fórach ako vývojári Unity a jeho dokumentácia je v niektorých bodoch nedotiahnutá. [7]

2.2 Unity 3D

Herný engine Unity vyvíja spoločnosť Unity Technologies a aktuálne je vo verzii 2019.1.2.

Prvá verzia bola predstavená v roku 2005 na Apple konferencii, zameranej na vývoj pre platformu OS X. Aktuálne Unity podporuje 27 platforiem, vrátane VR.

Najdôležitejšiu výhodou Unity je jednoduchosť v používaní. Vývoj v Unity je rýchly, a to špeciálne pre mobilné platformy. Výsledný program a projekt je v porovnaní s ostatnými enginami menší a tým výrazne uľahčuje export.

Unity má jednoduchú architektúru komponentov a skripty, písané v jazyku C#, sú pomerne rýchle a efektívne.

Veľkou výhodou je rozsiahla komunita vývojárov, ktorá poskytuje celé fóra rôznych odpovedí, rád a riešení na rôzne problémy, s ktorými sa vývojár môže stretnúť.

Taktiež Unity má vlastný komunitný trh, ktorý obsahuje kvantum pomerne lacných, prípadne bezplatných assetov.

Unity ale graficky zaostáva za Unreal Engine a CryEngine a má horšiu podporu pre terény a porast. [7]

2.3 CryEngine 5

Je vyvíjaný nemeckou spoločnosťou Crytek a bol predstavený v roku 2002.

Jedná sa o výkonný engine zameraný hlavne na platformu PC a herné konzoly.

CryEngine vyniká v realistickom osvetlení scény a jeho vykresľovacie možnosti sú veľmi vysoké spolu s pokročilou animáciou. Obsahuje výkonné nástroje na level dizajn a podporuje hustý porast terénu.

Jeho nevýhodou je, že je zložitý a nie je užívateľsky jednoduchý, takže sa nehodí pre začiatočníkov. [7]

2.4 Godot

Jedná sa o jediný open-source engine z vybraného zoznamu. Bol vytvorený komunitou a spadá pod licenciu MIT (Massachusetts Institute of Technology).

Jeho vývoj začal v už roku 2007, ale vydania sa dočkal až v roku 2014, keďže vývojári s ním neboli spokojní a neustále pridávali nové a nové možnosti a nástroje.

Je vytvorený v jazyku C a C++ a natívne podporuje C++ a C#. Podporuje aj niekoľko skriptovacích jazykov ako napríklad Rust alebo vlastný GDScript, podobný jazyku Python.

[8]

Tabuľka 1 Porovnanie herných enginov

Názov enginu	Jazyk jadra	Podporované jazyky	Rok vydania prvej verzie	Vydávajúca spoločnosť	Licencia a poplatky
Unreal Engine 4	C++	C++	1998	Epic Games	5% z výnosu
Unity	C++	C#	2005	Unity technologies	Mesačné poplatky na základe výnosu
Cry Engine	C++	C++, C#, Lua	2002	Crytek	5% z výnosu nad 5000 €
Godot	C/C++	C++, C#, Rust, GDScript	2014	Komunita	MIT licencia, zadarmo, open-source

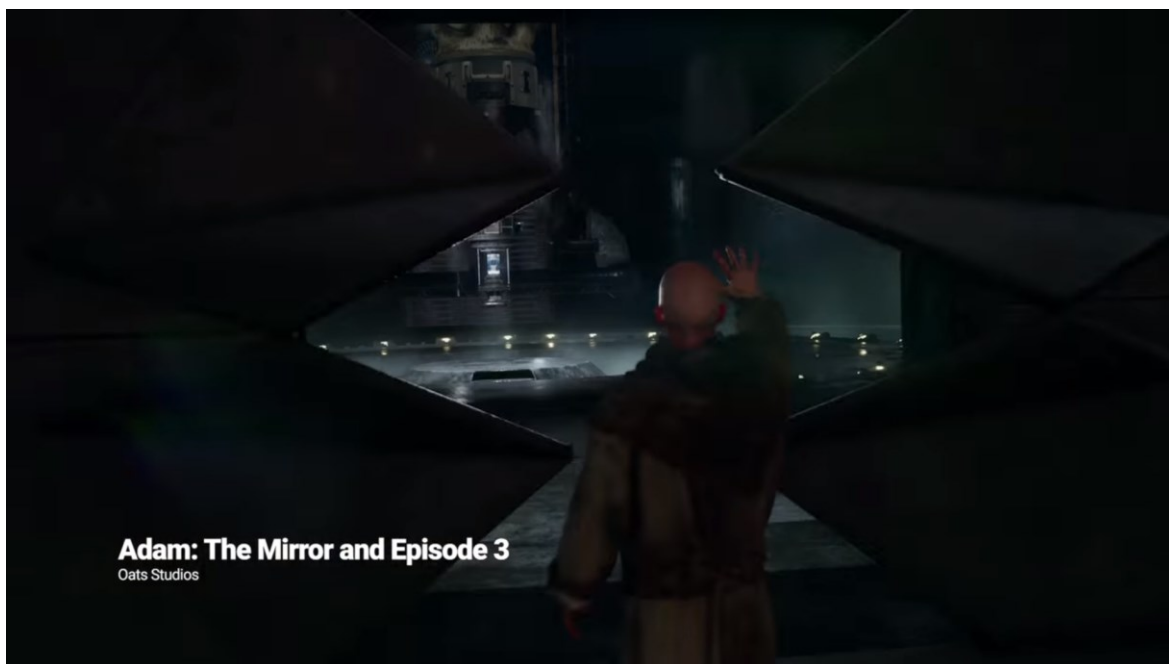
3 POUŽITĚ

Hoci herné 3D enginy boli primárne vyvinuté na tvorbu videohier, postupne sa začali objavovať možnosti ich využitia aj v iných odvetviach. Mnoho hier pôvodne vznikalo ako simulátor alebo editor a vice versa.

3.1 Komerčný priemysel

Komerčný, konkrétne herný priemysel pomohol ku vzniku herným enginom tak, ako ich poznáme dnes. Hry ale v komerčnom priemysle nie sú jediná možnosť, ako sa môžu tieto technológie uplatniť. Čoraz častejšie sa stretávame s interaktívnymi reklamami v rôznych podobách. Môže sa jednať o jednoduché 2D hry v podobe interaktívneho banneru, kompetitívne online hry v rámci kampane nejakej značky alebo prepracované ukážky vo virtuálnej alebo rozšírenej realite.

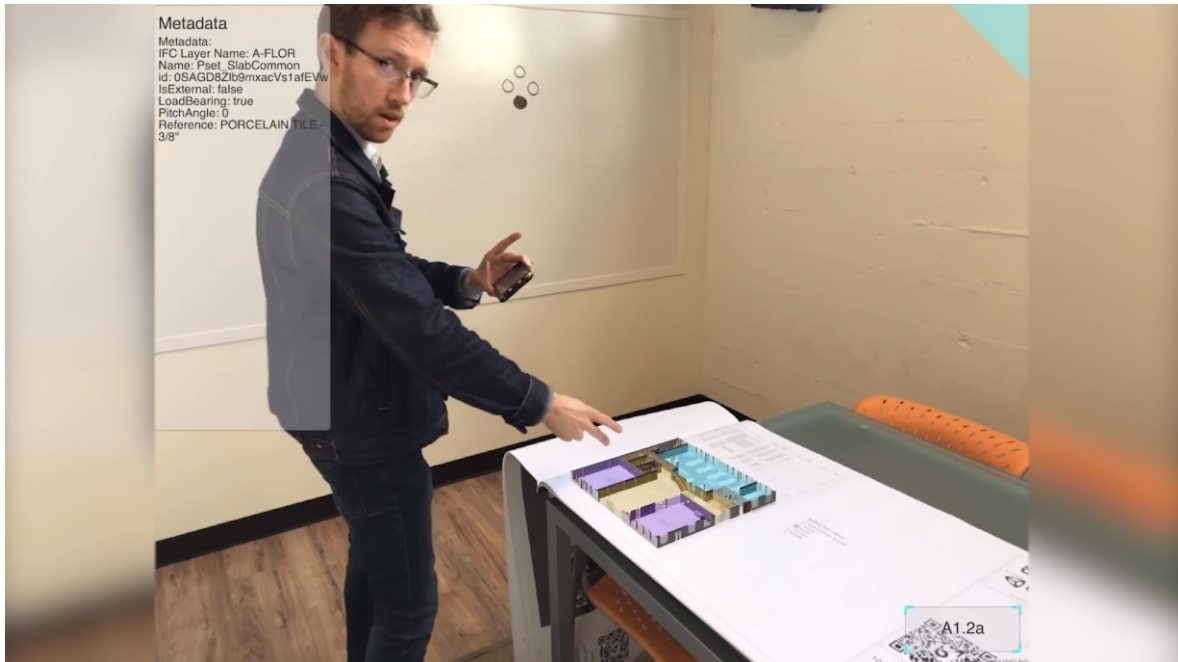
Ďalším rozmáhajúcim sa využitím herných enginov je tvorba animovaných filmov a filmov vo virtuálnej realite, kde môže naraz spolupracovať až niekoľko tímov. [4]



Obrázok 6 Animovaný film vytvorený v Unity

3.2 Technický priemysel

3D engine je ďalší z prostriedkov na vizualizáciu dát. Herný engine a virtuálna realita sa stáva silným nástrojom pre dizajnérov, inžinierov a konštruktérov, keďže klienti majú možnosť vidieť produkt skôr, než existuje a to bez fyzických obmedzení. [4]



Obrázok 7 Využitie rozšírenej reality v dizajne

3.3 Edukácia

Herný engine a virtuálna realita je silným nástrojom pre vizualizáciu informácií. Rôzne spoločnosti začínajú investovať do výukových aplikácií pre svojich zamestnancov, čím zvyšujú bezpečnosť a minimalizujú možné straty na majetku. Napríklad švédská spoločnosť Skanska, 5. najväčšia konštrukčná spoločnosť na svete, má vo virtuálnej realite kompletne bezpečnostné školenie pre svojich zamestnancov.



Obrázok 8 Edukačná aplikácia spoločnosti Skanska

Volkswagen Group má taktiež vlastnú výukovú aplikáciu, ktorá kompletne simuluje pracovné povinnosti zamestnanca vo výrobnjej hale. Vďaka tejto aplikácii môžu noví zamestnanci získať nutné vedomosti a schopnosti ešte predtým, než nastúpia na pracovisko a tak dokážu podať lepší výkon.



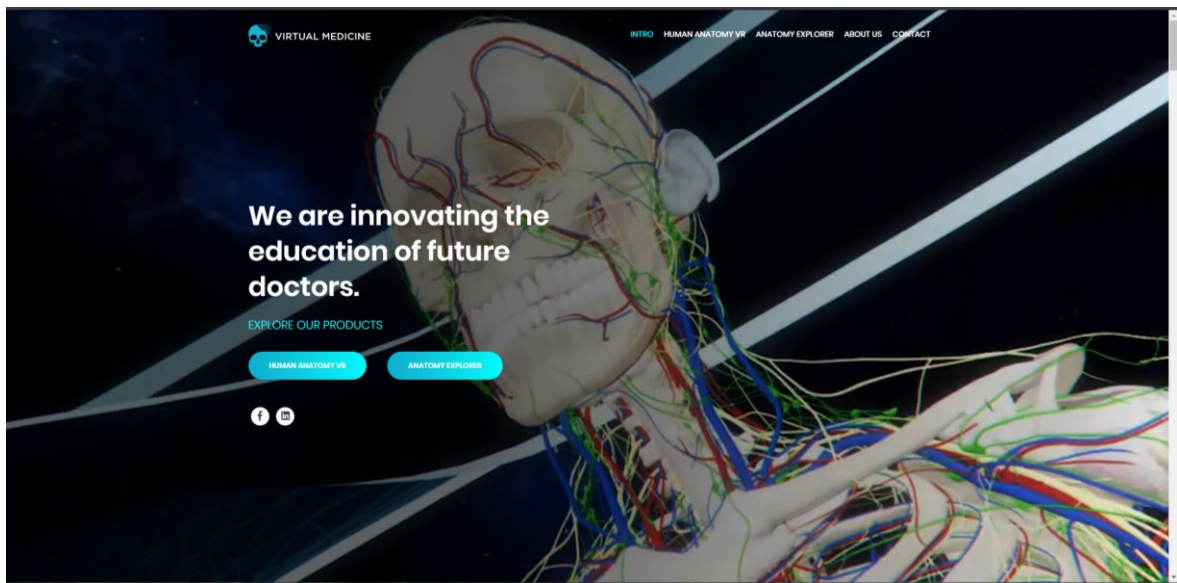
Obrázok 9 Tréningová aplikácia spoločnosti Volkswagen

Táto aplikácia sa používa na 120 výrobných miestach po celom svete. [4]

V roku 2017 vznikla na Slovensku aplikácia, ktorá pomáha študentom medicíny s anatómiou.

„Aplikácia Human Anatomy VR umožňuje vo virtuálnej realite vidieť jednotlivé štruktúry ľudského tela z rôznych perspektív v troch dimenziách. Študenti ich môžu interaktívne otáčať, nakláňať, zväčšovať či oddeľovať tak, akoby mali reálne sústavy ľudského tela pred sebou. Aplikáciu, ktorá obsahuje aj prvky gamifikácie a umožňuje tak hrovou a interaktívnou formou otestovať svoje znalosti, vytvoril slovenský startup Virtual Medicine. Verzia aplikácie pre stredné školy je upravená, aby zodpovedala osnovám stredoškolského učiva, je dostupná v slovenčine a latinčine a obsahuje všetky zložky ľudského tela, teda svaly a kosti, cievy, nervy, orgány či jednotlivé sústavy.“

„Technologickým partnerom projektu je spoločnosť Samsung Electronics, ktorá škole venovala vybavenie pozostávajúce z desiatich setov pre virtuálnu realitu (Samsung Galaxy S8 a Samsung Gear VR) a Smart signage interaktívnej obrazovky s výkonným operačným systémom TIZEN. Celkové náklady na zriadenie virtuálnej učebne sa vyšplhali na takmer 40-tisíc eur. Virtual Medicine bude projekty na školách priebežne vyhodnocovať a chce osloviť aj ďalšie stredné a vysoké školy. Nielen na Slovensku, ale aj v zahraničí.“ [5]



Obrázok 10 Virtual Medicine

3.4 Armáda

Vďaka pokročilým fyzikálnym enginom sú herné enginy využívané aj na armádne simulácie a tréningy. Tieto aplikácie sú v armáde označované ako seriózne hry. Prvá seriózna hra vznikla v roku 1980 na požiadavku armády USA a jednalo sa o modifikáciu hry Battlezone od Atari. Od tejto doby využíva armáda USA seriózne hry na tréning spolupráce medzi vojakmi, simulácie pre postup zložitým terénom a taktiež na nábor nových regrútov. [6]

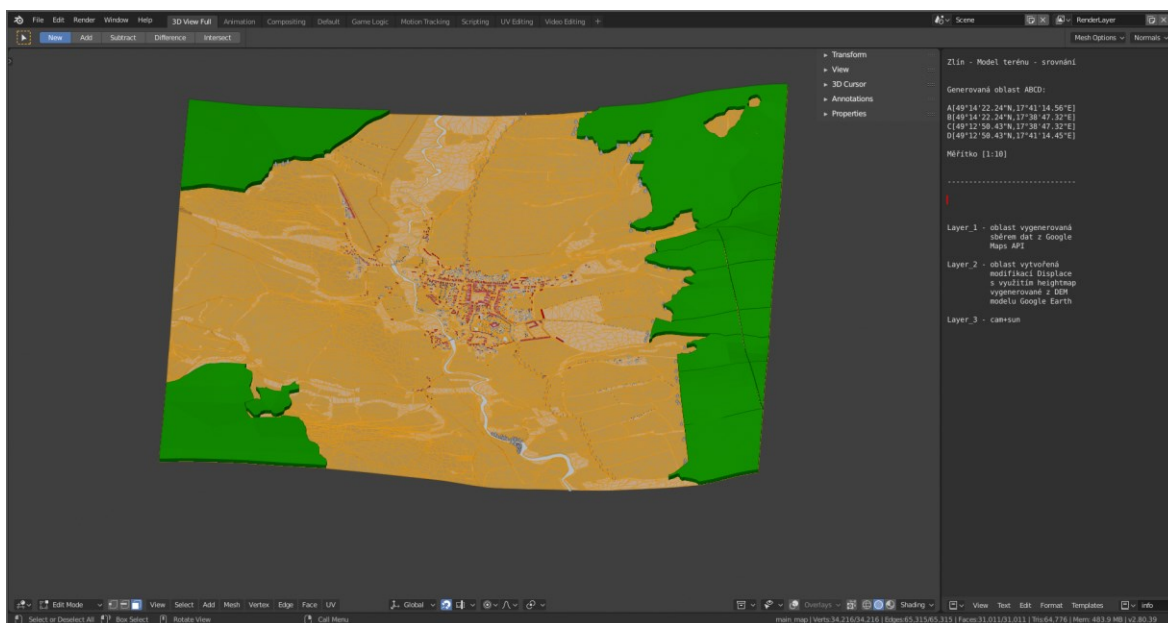


Obrázok 11 Seriózna hra pre armádu USA

II. PRAKTICKÁ ČASŤ

4 VÝBER VHODNÉHO PROJEKTU PRE DEMONŠTRÁCIU

Ako projekt pre demonštráciu som vybral prekonvertovanie modelu mesta Zlín z roku 1918, a to na podnet môjho vedúceho bakalárskej práce, ktorý tento model mal k dispozícii. Model bol vytvorený v programe Blender. Model bolo možné prehliadať cez webové rozhranie za pomoci rozšírenia Blender4Web, využívajúceho technológie HTML5 a WebGL. Kvôli obrovskému rozsahu scény bolo prehliadanie modelu náročné na hardware a danú technológiu a tak vznikla myšlienka prekonvertovania tohto modelu do herného engine, ktorý je presne na podobné účely určený. Model obsahoval terén vygenerovaný z Google Maps API, pre čo najvyššiu autenticitu. Tento námet ma zaujal, keďže sa jedná o projekt, ktorý perfektne využije vlastnosti a technológie herného engine, ktorý je na podobné účely dizajnovaný.



Obrázok 12 Model v programe Blender

5 VÝBER VHODNÉHO 3D ENGINU

Napriek tomu, že väčšina dostupných enginov ponúka pomerne rovnakú funkcionálnosť, niektoré projekty môžu vyžadovať špecifickú funkcionálnosť, ktorá je kvalitnejšie implementovaná v jednom konkrétnom engine. Môže sa jednať o to, ako zvláda veľké scény alebo možnosti prístupu viacerých užívateľov naraz. Preto je potrebné si urobiť prieskum v počiatku projektu, keďže kompatibilita medzi jednotlivými engineami je minimálna a nutná zmena engineu v priebehu vývoja často znamená veľké komplikácie.

V tomto odseku zamerám na svoj postup výberu rozdelený do bodov.

Pokiaľ sa jedná o komerčný projekt, je jasné, že veľkým kritériom je cena a licenčné podmienky.

5.1 Programovací jazyk

V hernom vývoji sa dlhodobo používa primárne C++ vďaka jeho rýchlosti, keďže sa jedná o nižší programovací jazyk v porovnaní s ostatnými objektovo orientovanými jazykmi.

C++ je primárny programovací jazyk väčšiny enginov, až na Unity, ktorý používa striktné C#.

Ja som sa rozhodol pre C++ , keďže ho osobne preferujem nad C# a mám s ním rozsiahlejšie skúsenosti.

Unreal Engine 4 má taktiež aj svoj grafický jazyk nazývaný blueprint, ktorý výrazne zjednodušuje prototypovanie, urýchľuje a sprehľadňuje vývoj.

5.2 Prostredie

Podobne, ako iné vývojové prostredia, majú herné enginy možnosti modulovania, preskupovania a otvárania nových okien a nástrojov. Najprehľadnejší a najefektívnejší mi v tomto prípade Unity, keďže má veľmi prehľadný prehliadač obsahu a editor objektov. O niečo menej je prehľadný Unreal Engine 4, ktorý to kompenzuje vstavaným editorom 3D objektov.

5.3 Osobný dojem

Na mňa najviac zapôsobil Unreal Engine 4, hoci som mal v minulosti viac skúseností s Unity, prechod na Unreal Engine 4 bol pre mňa veľmi jednoduchý a výrazne mi zjednodušil prácu. Veľkou výhodou pre mňa je, že pri riešení jednoduchších problémov a algoritmov

nemusím prepínať medzi vývojovým prostredím pre program a vývojovým prostredím enginu, ale dokážem potrebný algoritmus navrhnuť, otestovať a odladiť priamo na jednom mieste. Ďalším plusom je, že sa jedná o grafické programovanie, takže samotný kód je aj diagramom, čo mi dáva výborný prehľad o toku dát medzi jednotlivými metódami a triedami. Jednotlivé bloky sa dajú prehliadať v katalógu, takže nie je nutné detailne ovládať všetky dostupné metódy a funkcie a ešte k tomu každá položka obsahuje aj krátky popis funkčnosti, čo výrazne pomáha začínajúcim vývojárom s oboznámením sa s funkciami a možnosťami enginu.

5.4 Výsledok

Na základe technických parametrov porovnávaných enginov a požiadaviek zvoleného projektu, som vybral ako vhodný engine Unreal Engine 4. Tento engine sa často využíva v hrách a aplikáciách, ktoré obsahujú otvorený svet veľkých rozmerov a s vysokým počtom objektov. V tomto engine je možné vytvoriť dokonca celé svety vďaka technológiám ako sublevely a prednastaveným optimalizáciám ako inšancovanie statických meshov a vykresľovaniu iba viditeľných objektov. Taktiež v tomto engine je jednoduché vytvoriť hraciu postavu s priradenou kamerou pomocou grafického programovania blueprint.

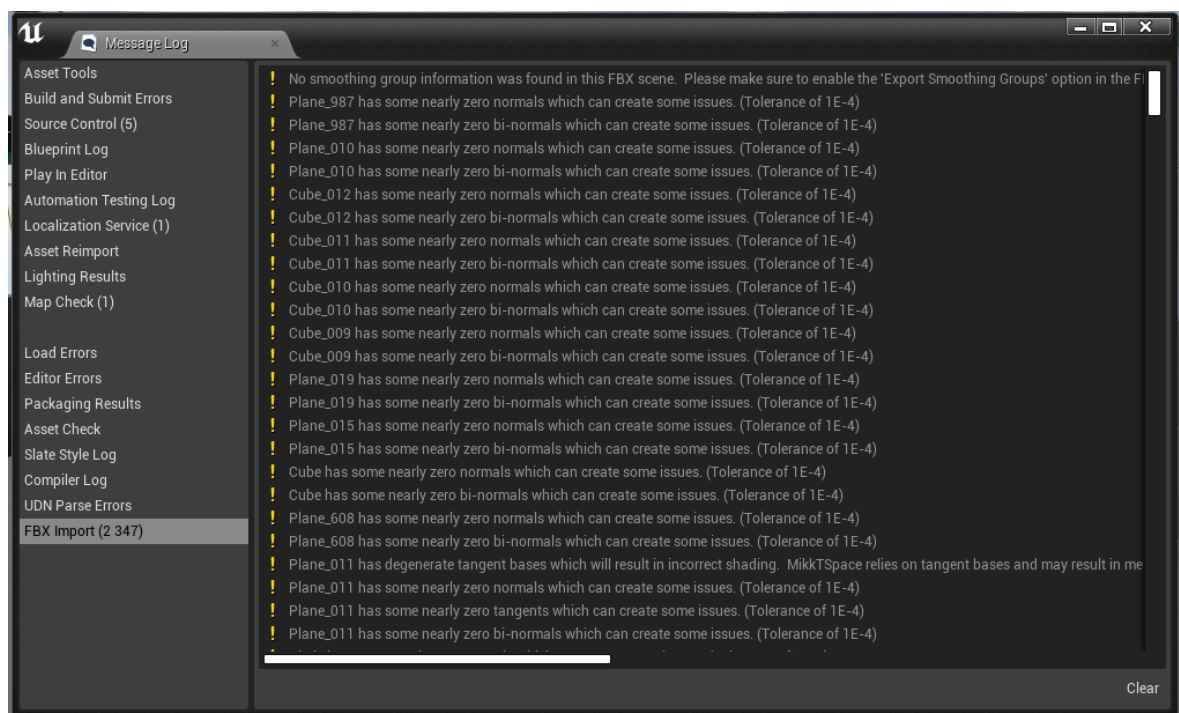
6 POSTUP

6.1 Import Scény do enginu

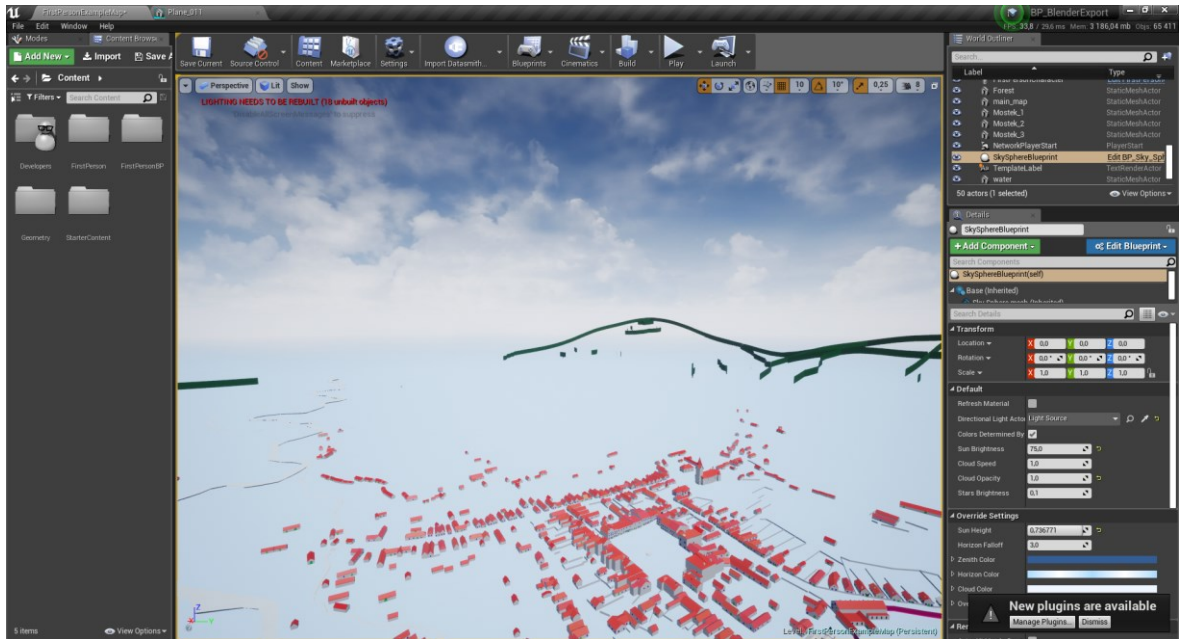
Prvým krokom pri konverzii z programu Blender bol export kompletnej scény do formátu, ktorý je obojstranne podporovaný. Na výber boli 2 možnosti, a to formát .obj a .fbx. Formát .obj je vhodnejší na export jednotlivých objektov a tvarov, no zároveň nedokáže exportovať objekt aj s jeho materiálom. Preto som zvolil formát .fbx.

V tomto bode nastala prvá komplikácia, keďže Blender oficiálne nepodporuje export do Unreal Engine 4 a vice-versa. Hlavné problémy, ktoré nastali, bola mierka, keďže Unreal Engine 4 využíva ako svoju natívnu jednotku centimeter a Blender má svoju vlastnú jednotu blender unit.

Ďalej to boli nekompatibilné a nekvalitné normálové mapy. Blender je v tomto ohľade veľmi tolerantný a dokáže vykresliť aj objekty s veľmi nekvalitnou normálovou mapou, kdežto herný engine vyžaduje vyššiu kvalitu vstupných dát. Najviac sa to ukázalo na samotnom vygenerovanom teréne.



Obrázok 13 Výpis chýb po importe do enginu



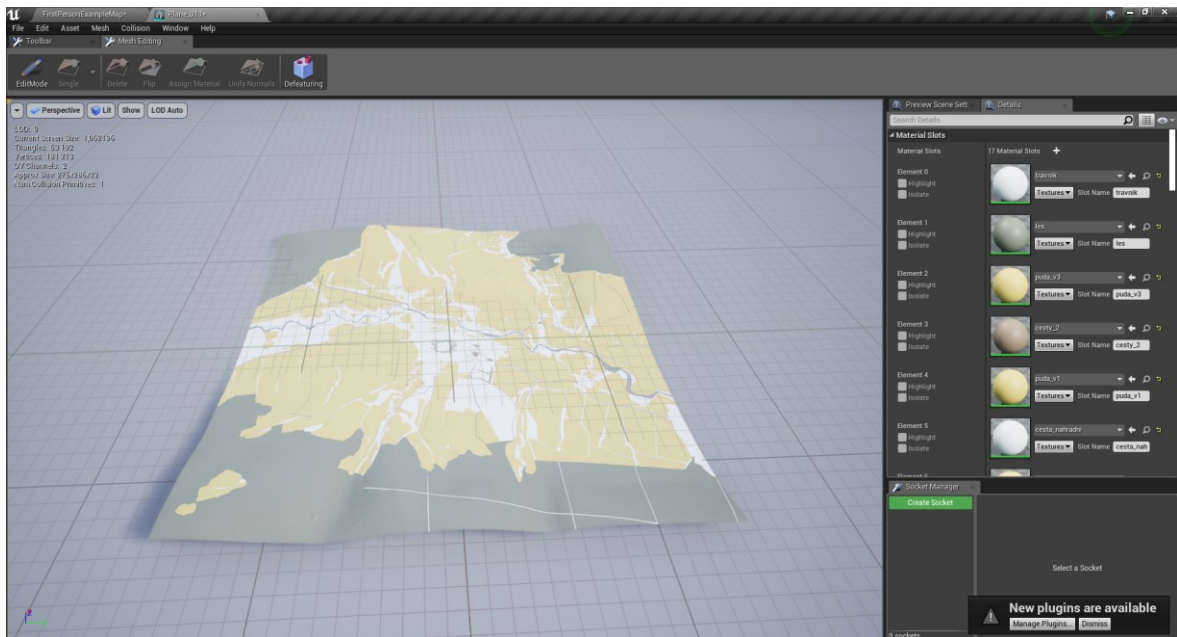
Obrázok 14 Ukážka scény po importe

Siahol som preto po neoficiálnom rozšírení pre export z Blenderu do Unreal Engine 4. Žiaľ výsledok nebol dostatočne uspokojivý.

Ďalšou možnosťou bol export do iného grafického modelovacieho prostredia, ktorý priamo podporuje export do Unreal Engine 4, čo bol v mojom prípade Maya Lite, priamo určený pre herných vývojárov. Výsledok bol opäť neuspokojivý, keďže po exporte sa prejavili to- tožné chyby ako pri exporte do enginu.

Keďže mám rozsiahlejšie skúsenosti s prácou priamo v hernom engine, ako s prácou v 3D modelovacom prostredí, rozhodol som sa, že využijem dostupné nástroje pre úpravu 3D objektov, ktoré sú implementované priamo v hernom engine.

Unreal Engine 4 poskytuje editor statických meshov, ktorý umožňuje upravovať jednotlivé vertexy a prácu s orientáciou normál. V tomto nástroji som opravil jednotlivé objekty, aby mali správnu orientáciu normály a vykresľovali sa korektne.



Obrázok 15 Opravené normály terénu

6.2 Úprava scény v hernom engine

Pri úprave jednotlivých objektov som si potvrdil, že nástroje herného engine sú skoro rovnako účinné, ako tie v modelovacom prostredí.

Po prácnej úprave väčšiny objektov na scéne som prešiel k priradovaniu materiálov k jednotlivým objektom a ich častiam. Tu som narazil na problém so zakrivenou plochou, vygenerovanou z údajov z Google Maps, slúžiacou ako môj terén. Keďže táto plocha bola automaticky generovaná, jednotlivé úseky, ktoré oddeľujú rôzne typy materiálov objektu, konkrétne pri teréne sa jedná o cesty, trávnaté plochy a mestskú dlažbu, nevyzerali vôbec prirodzene a nemali rovnakú mierku. Výsledkom toho bolo, že po priradení materiálu viacerým úsekom rovnakého typu bolo nutné vytvoriť novú inštanciu materiálu pre každý geometrický úsek, túto inštanciu rotovať a škálovať, aby sa zhodovala so susednými. Toto riešenie bolo veľmi neefektívne.

Alternatívou k tomuto problému bolo vygenerovanie nového objektu, priamo typu terén, ktorý obsahuje engine. Objekt tohto typu je možné ďalej tvarovať a upravovať nástrojmi, ktoré engine obsahuje a taktiež je možné naň kresliť rôznymi materiálmi v niekoľkých vrstvách, ktoré sa môžu prelínať a tým vytvárajú realistický dojem. Engine obsahuje možnosť generovania terénu z .png súboru, takzvanej výškovej mapy. Podarilo sa mi získať pôvodnú výškovú mapu, ktorá bola použitá pre generovanie terénu v programe Blender.

Výsledok vygenerovania terénu nebol stopercentne uspokojivý, keďže nový terén sa nie úplne zhoduje s pôvodným a niektoré objekty sa ocitli mimo, alebo v teréne.

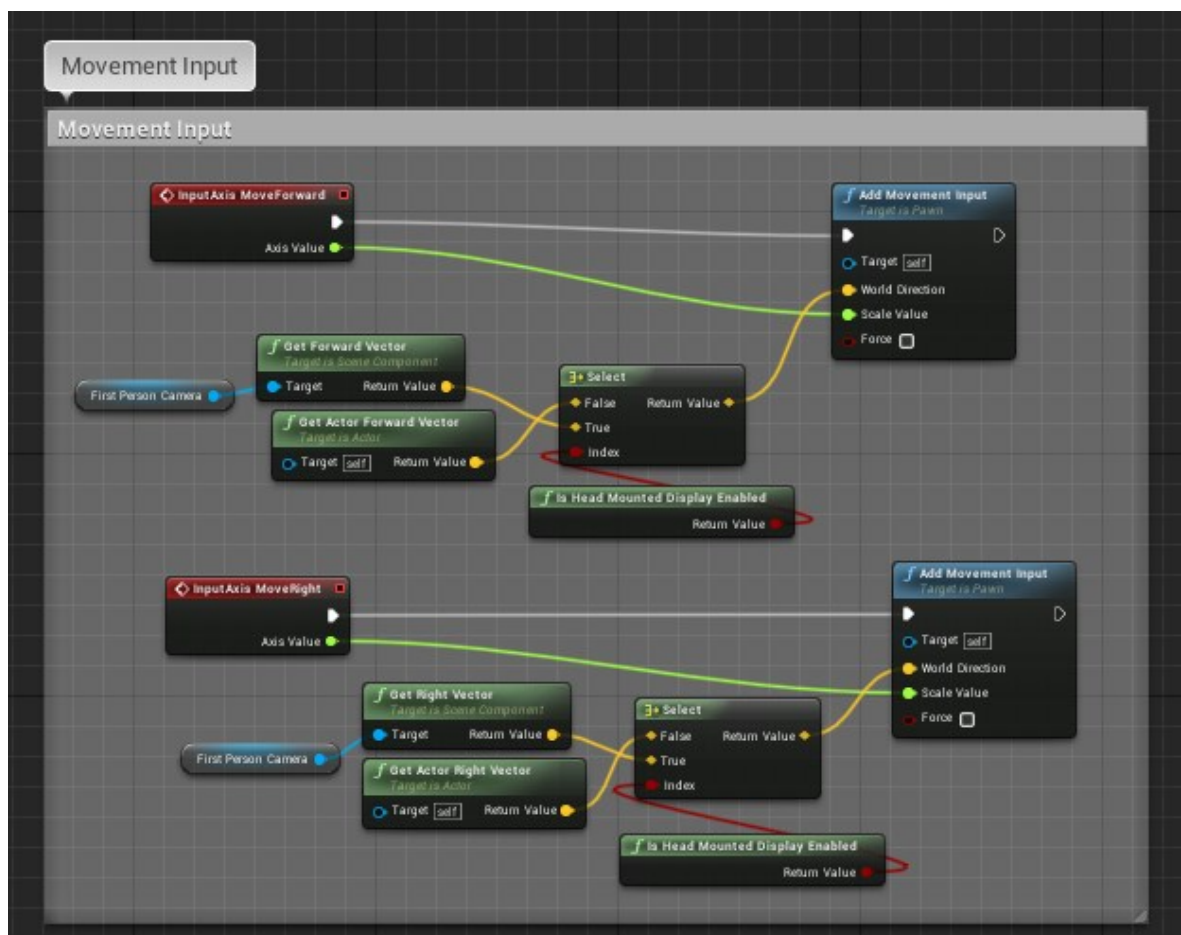
6.3 Kamera a postava hráča

Pre svoj projekt som sa rozhodol pre kameru z prvej osoby.

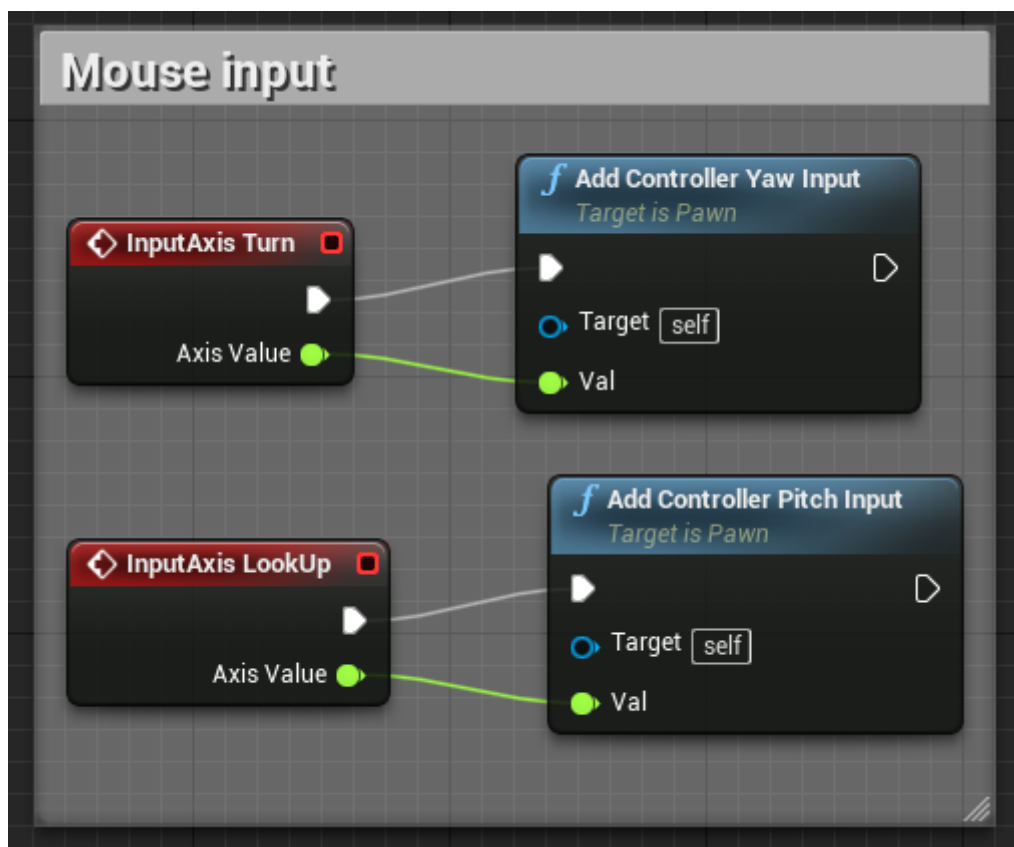
Ovládanie pohybu hráča bude pomocou kláves W,A,S a D a skok pomocou medzerníka.

Celú funkcionálnosť som naprogramoval pomocou grafického programovacieho jazyka blueprint.

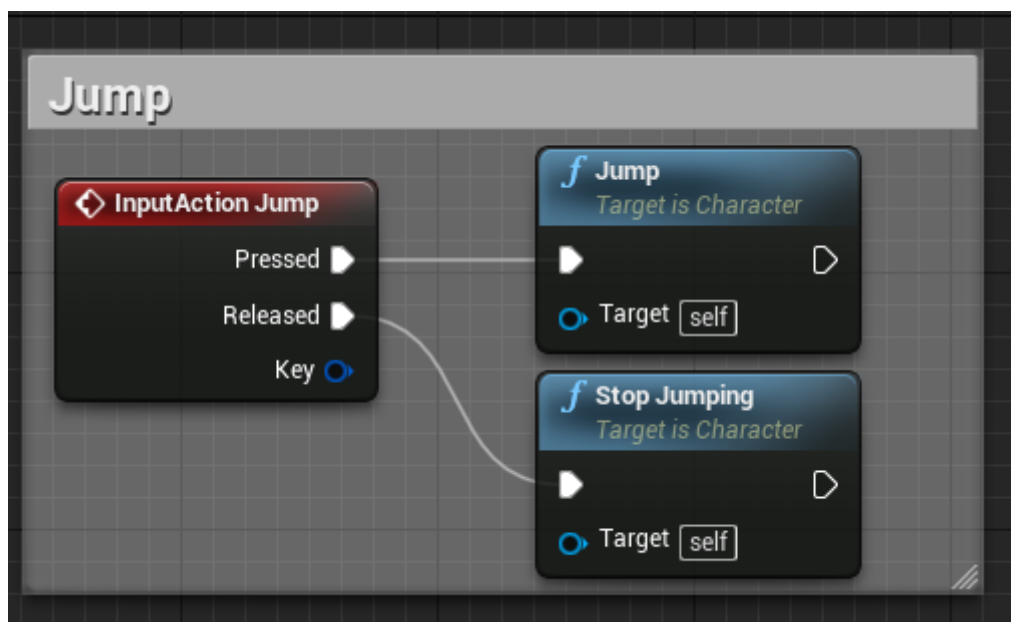
Program som rozdelil do 3 častí, ktoré sú: input pohybu, input myši a skok.



Obrázok 16 Input pohybu



Obrázok 17 Input myši



Obrázok 18 Skok

Materiály

Materiály čerpám z knižnice fotorealistických materiálov Substance, ktorý poskytuje študentskú licenciu, inak sa za jednotlivé materiály platí. Síce je študentský prístup obmedzený, zato ale pod študentskou licenciou je neobmedzený prístup na komunitný trh Substance share, kde som našiel všetky potrebné textúry a materiály.

Keďže terén obsahuje niekoľko rôznych typov povrchov, bolo nutné si vytvoriť vrstvený materiál, ktorý som potom priradil terénu.

Jednotlivé vrstvy je možné potom do seba miešať pomocou štetca a vytvoriť tak realistický dojem.

6.4 Optimalizácia

Aplikácia po importe bola veľmi náročná na hardware, keďže vygenerovaný terén je veľmi zložitý a rozsiahly a v scéne sa nachádza kvantum rovnakých, ale samostatných objektov, čo je dôsledok toho, že pôvodná scéna vytvorená v 3D programe nevyužívala inšancovanie, alebo ho nebolo možné exportovať, takže do pamäti GPU sa načíta rovnaký objekt v plnej veľkosti niekoľkokrát. Pri inštancii by v pamäti boli uložené len jeho transformačné údaje a poloha na scéne.

Riešením je odstránenie rovnakých objektov, ako zo scény, tak aj z balíčku a nahradenie ich inštanciami jedného objektu. Ďalší krok je obmedzenie viditeľnosti scény z pohľadu hráča a zredukovanie členitosti terénu pomocou nástrojov enginu.

6.5 Výsledok a zhodnotenie

Z problémov, ktoré nastali v mojej práci vyplýva, že kompletný export scény z 3D programu do herného enginu nie je optimálny proces, keďže môže dôjsť k inkompatibilite medzi mierkami a orientáciami dvoch rôznych prostredí. Ďalšou vecou je, že model nemusí obsahovať inšancovanie, ako tomu bolo v modeli, ktorý slúžil ako podklad pre moju prácu, čo je pre herný engine extrémna záťaž, keďže sa v jednom leveli nachádzajú stovky samostatných objektov. Úprava a oprava exportovanej scény pravdepodobne zabrala viac času, ako by zabralo znovuvytvorenie celej scény v hernom engine. V praxi to funguje tak, že scény sa vytvárajú až v hernom engine a grafici poskytujú assety level dizajnérom, ktorý túto scénu modelujú.

7 MOŽNOSTI PRE DALŠÍ ROZVOJ

7.1 Vytvorenie scény v hernom engine

Ako som spomínal v predošlej kapitole, nutné riešenie pre ďalší rozvoj je využitie assetov z pôvodného 3D modelu a kompletne znovuvytvorenie scény už priamo v hernom engine, s ohľadom na to, ako funguje herný engine a jeho komponenty. Tým by sa odstránili všetky inkompatibility a chyby z exportu a výrazne by sa zvýšil výkon aplikácie vďaka inšancovaniu objektov.

7.2 Virtuálna a rozšírená realita

Pridanie podpory virtuálnej alebo rozšírenej reality by dodalo práci nový rozmer, keďže by bolo možné priamo porovnať a zažiť, akými zmenami prešlo mesto Zlín za posledných sto rokov. Takáto aplikácia by mohla nájsť svoje uplatnenie v múzeách a v školách.

7.3 Optimalizácia

Aplikácia je v aktuálnom stave veľmi náročná na hardware. Riešením je odstránenie duplicitných objektov a nahradenie ich inštanciami jedného objektu.

7.4 Náplň

Scéna by mohla perfektne slúžiť ako hra alebo edukačná aplikácia, keďže je modelom celého mesta. Edukačná aplikácia by mohla pri každej zaujímavej budove alebo mieste zobrazit' jeho históriu, alebo by simulovala život v danej dobe.

7.5 Uživatelské rozhranie

V aplikácii nie je žiadne užívateľské rozhranie alebo menu, čo vidím ako prvý a najhlavnejší bod pre rozvinutie aplikácie, keďže užívateľské rozhranie by mala obsahovať každá plnohodnotná aplikácia.

ZÁVER

Hlavný cieľom tejto práce bolo preskúmanie a popis herných enginov a ich možností, ktoré som demonštroval v praktickej časti svojej práce. Cieľom praktickej časti bol export veľkej scény z 3D programu Blender a následný rozbor prostredia aplikácie, ktoré vzniklo takýmto spôsobom. Základom pre moju prácu bol model mesta Zlín z roku 1918, ktorý mi poskytol vedúci mojej bakalárskej práce, no bohužiaľ model nebol vytvorený s vedomím toho, že by v budúcnosti mohol byť využitý ako podklad pre aplikáciu v hernom engine a pri použití v hernom engine je veľmi výkonnostne náročný. Pre funkčnosť bolo nutné model prispôbiť a upraviť pre potrebu herného enginu a tieto opravy som sa rozhodol vykonať priamo v hernom engine, aby som demonštroval jeho možnosti a schopnosti.

Tieto nedostatky spôsobili obrovskú stratu času, ktorý mohol byť vynaložený na tvorbu funkcionality, obsahu a používateľského rozhrania aplikácie.

Výsledkom práce je model mesta, ktorý je možné prechádzať z pohľadu prvej osoby, ktorý je pripravený na implementáciu virtuálnej alebo rozšírenej reality.

Dôvodom vzniku tejto aplikácie je sprístupnenie modelu mesta bežným ľuďom, ktorý neovládajú prácu v 3D programe.

Vytvorená aplikácie môže byť ďalej upravovaná, vylepšovaná, keďže Unreal Engine 4 je bohatý na funkcionality.

ZOZNAM POUŽITEJ LITERATURY

- [1] OKITA, Alex. *Learning C# programming with Unity 3D*. Second edition. Boca Raton, FL: CRC Press/Taylor & Francis Group, 2019. ISBN 9781138336810.
- [2] Unreal Engine 4 Documentation [online]. [cit. 2019-05-12]. Dostupné z: <https://docs.unrealengine.com>
- [3] BUSBY, Jason, Zak PARRISH a Jeff WILSON. *Mastering Unreal Technology, Volume I: Introduction to Level Design with Unreal Engine 3*. Pearson Education, 2009. ISBN 9780768688979.
- [4] *Real-time solutions. Endless opportunities.: Games, Film, Automotive, AEC, Gambling, and more. Forged in gaming, Unity's flexible real-time tools offer incredible possibilities for all industries and applications.* [online]. [cit. 2019-05-12]. Dostupné z: <https://unity.com/solutions>
- [5] KADLEC, Roman. Anatomia sa vo virtuálnej realite učí už aj na slovenských stredných školách. *TOUCHIT* [online]. 16.11. 2017 [cit. 2019-05-12]. Dostupné z: <https://touchit.sk/anatomia-sa-vo-virtualnej-realite-uci-uz-aj-na-slovenskych-strednych-skolach>
- [6] SMITH, Peter et al. *Serious Games and Their Use in NATO* [online]. 2013 [cit. 2019-05-12]. ISBN 9789283702047. Dostupné z: <https://www.sto.nato.int/publications/>
- [7] ŠMÍD, Antonín. *Comparison of Unity and Unreal Engine* [online]. Praha, 2017 [cit. 2019-05-12]. Dostupné z: <https://dcgi.fel.cvut.cz/theses/2017/smidanto>. Bakalárska Práca. ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE. Vedoucí práce Jiří Bitner.
- [8] *Godot Engine: Free and open source 2D and 3D engine* [online]. [cit. 2019-05-12]. Dostupné z: <https://godotengine.org/>
- [9] LEE, Joanna. *Learning Unreal Engine Game Development*. Packt Publishing, 2016. ISBN 978-1784398156.

ZOZNAM POUŽITÝCH SYMBOLOV A SKRATIEK

GPU Grafický procesor, anglicky Graphics Processing Unit .

IDE Integrované vývojové prostredie, anglicky Integrated Development Envirment.

RAM Random Access Memory.

PC Osobný počítač, anglicky Personal Computer

ZOZNAM OBRÁZKOV

Obrázok 1 Pseudo-3D hra	11
Obrázok 2 Ukážka prostredia Unreal Engine 4	13
Obrázok 3 Komponenty herného enginu	16
Obrázok 4 Herná slučka.....	18
Obrázok 5 Herná slučka založená na udalostiach.....	19
Obrázok 6 Animovaný film vytvorený v Unity.....	25
Obrázok 7 Využitie rozšírenej reality v dizajne	26
Obrázok 8 Edukačná aplikácia spoločnosti Skanska	27
Obrázok 9 Tréningová aplikácia spoločnosti Volkswagen.....	28
Obrázok 10 Virtual Medicine	29
Obrázok 11 Seriózna hra pre armádu USA	30
Obrázok 12 Model v programe Blender	32
Obrázok 13 Výpis chýb po importe do enginu	35
Obrázok 14 Ukážka scény po importe	36
Obrázok 15 Opravené normály terénu.....	37
Obrázok 16 Input pohybu	38
Obrázok 17 Input myši	39
Obrázok 18 Skok	39

ZOZNAM TABULIEK

Tabuľka 1 Porovnanie herných enginov	24
--	----

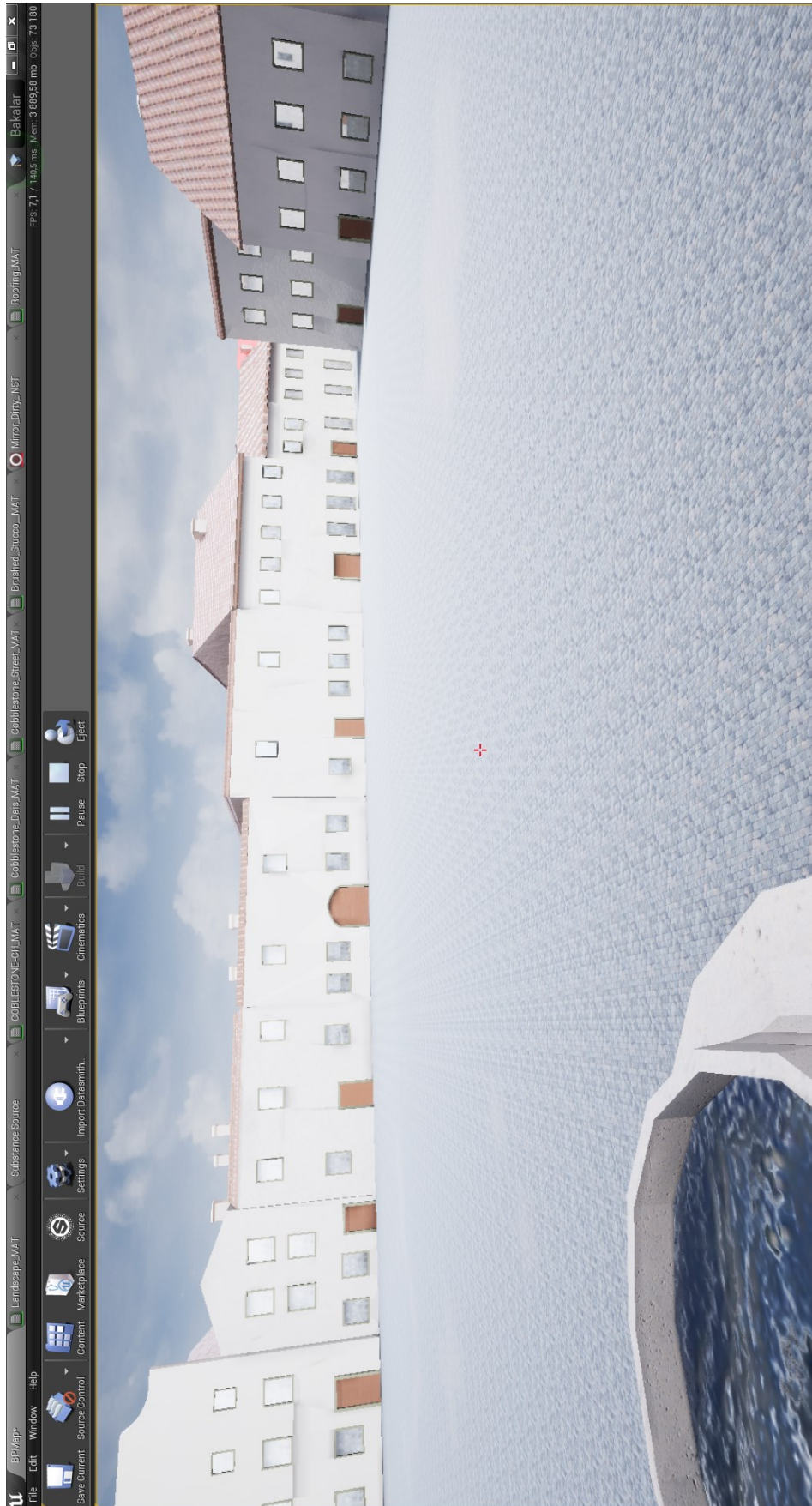
ZOZNÁM PRÍLOH

PI Námestie

PII Radnica

PIII Materiál terénu

PRÍLOHA P I: NÁMESTIE



PRÍLOHA P II: RADNICA



PRÍLOHA PIII: MATERIÁL TERÉNU

