

CryEngine a jeho praktické využití

CryEngine and Its Practical Use

Roman Mareček

Bakalářská práce
2023

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Roman Mareček
Osobní číslo: A20073
Studijní program: B0613A140020 Softwarové inženýrství
Forma studia: Kombinovaná
Téma práce: CryEngine a jeho praktické využití
Téma práce anglicky: CryEngine and Its Practical Use

Zásady pro vypracování

1. Podrobně se seznamte s nástrojem CryEngine.
2. V práci popište jeho hlavní vlastnosti a možnosti.
3. Vytvořte podrobnou dokumentaci pro CryEngine v českém jazyce.
4. Navrhněte a vytvořte komplexní aplikaci, která bude demonstrovat širokou škálu funkcí CryEngine.
5. Vytvořenou aplikaci doplňte podrobným popisem její tvorby.

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

Seznam doporučené literatury:

1. MARCOUX III, Richard Gerard, Chris GOODSWEN, Riham TOULAN a Sam HOWELS. CRYENGINE Game Development Blueprints. 1. Birmingham B3 2PB, UK.: Packt Publishing, 2015. ISBN 9781784399870.
2. GUNDLACH, Sascha a Michelle K. MARTIN. Mastering CryENGINE. 1. Birmingham B3 2PB, UK.: Packt Publishing, 2014. ISBN 9781783550258.
3. CRYENGINE V Manual. CRYENGINE V Manual [online]. [cit. 2022-11-22]. Dostupné z: <https://docs.cryengine.com/display/CEMANUAL/CRYENGINE+V+Manual>
4. Tutorials. Tutorials [online]. [cit. 2022-11-22]. Dostupné z: <https://www.cryengine.com/tutorials>
5. GAMEDEV4FUN. CryEngine 5 Tutorials. CryEngine 5 Tutorials [online]. [cit. 2022-11-22]. Dostupné z: <https://www.youtube.com/playlist?list=PLYnbKITbqv7BgLbTynsBwmOfk8pfiGpei>

Vedoucí bakalářské práce:

Ing. Pavel Pokorný, Ph.D.

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce: **2. prosince 2022**

Termín odevzdání bakalářské práce: **26. května 2023**



doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 7. prosince 2022

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 bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- 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 – 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

Roman Mareček, v.r.
podpis studenta

ABSTRAKT

Hlavním cílem bakalářské práce je popsat a představit hlavní vlastnosti a možnosti vývojového prostředí CryEngine (verze 5.7.1 LTS), vytvořit dokumentaci v českém jazyce a vyvinout komplexní aplikaci, ve které bude představena široká škála funkcí a možností CryEngine. Teoretická část práce je zaměřena na popis a vlastnosti daného engine. Praktická část se zaměřuje na použití co největšího množství nástrojů, které CryEngine nabízí. První část praktické části práce je zaměřena na tvorbu dokumentace v českém jazyce. Druhá část praktické části práce je pak zaměřena na tvorbu aplikace s popsányými kroky tvorby. Hlavním výstupem této bakalářské práce je aplikace, která demonstruje širokou škálu funkcí tohoto engine a manuál v českém jazyce k engine CryEngine.

Klíčová slova: CryEngine, dokumentace, herní engine

ABSTRACT

The main goal of the bachelor thesis is to describe and present the main features and possibilities of the CryEngine development environment (version 5.7.1 LTS), to create documentation in Czech language and to develop a comprehensive application that will present a wide range of CryEngine features and possibilities. The theoretical part of the thesis is focused on the description and features of the engine. The practical part focuses on using as many tools as possible that CryEngine offers. The first part of the practical part of the thesis is focused on creating documentation in Czech language. The second part of the practical part of the thesis is focused on creating an application with described steps of creation. The main output of this bachelor thesis is an application that demonstrates a wide range of features of this engine and a manual in Czech language for the CryEngine engine.

Keywords: CryEngine, documentation, gaming engine

Děkuji vedoucímu své bakalářské práce doktorovi Ing. Pavlu Pokornému, Ph.D. za pomoc, nápady a podněty k této bakalářské práci. Rovněž chci poděkovat svým rodičům, spolužákům a přátelům za podporu a pomoc při studiu.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

| | |
|--|-----------|
| ÚVOD | 9 |
| I TEORETICKÁ ČÁST | 10 |
| 1 PŘEDSTAVENÍ A SEZNÁMENÍ S CRYENGINE | 11 |
| 2 HLAVNÍ VLASTNOSTI A MOŽNOSTI | 12 |
| 2.1 VIZUALIZACE | 12 |
| 2.1.1 Oblastní světla | 12 |
| 2.1.2 Podpora DirectX 12..... | 13 |
| 2.1.3 Fyzikální vykreslování | 13 |
| 2.1.4 Dynamická vodní kaustika v reálném čase | 13 |
| 2.1.5 Teselace..... | 14 |
| 2.1.6 3D HDR odlesky objektivu | 15 |
| 2.1.7 Efektivní antialiasing | 15 |
| 2.1.8 Rozmazání pohybu a hloubka ostrosti | 16 |
| 2.1.9 Místní reflexe v reálném čase | 17 |
| 2.1.10 Realistická vegetace | 17 |
| 2.1.11 Globální osvětlení založené na voxelech | 18 |
| 2.1.12 Mapy stínů pro jednotlivé objekty | 18 |
| 2.1.13 Směrová okluze v prostoru obrazovky..... | 19 |
| 2.1.14 Filmové mapování tónů HDR | 20 |
| 2.1.15 Osvětlení na základě obrazu..... | 20 |
| 2.1.16 Systém částicového efektu | 21 |
| 2.1.17 Objemové stíny mlhy | 21 |
| 2.1.18 Mapování paralaxní okluze | 22 |
| 2.2 SANDBOX | 22 |
| 2.2.1 CryEngine Sandbox | 22 |
| 2.2.2 Editor materiálů..... | 22 |
| 2.2.3 Podpora FBX..... | 23 |
| 2.2.4 Intuitivní design úrovní | 23 |
| 2.2.5 Filmový editor Trackview..... | 24 |
| 2.2.6 Flowgraph | 24 |
| 2.2.7 Integrace látek | 25 |
| 2.3 UMĚLÁ INTELIGENCE A ANIMACE | 26 |
| 2.3.1 Technologie postavy | 26 |
| 2.3.2 Parametrická skeletová animace | 26 |
| 2.3.3 Geometrická mezipaměť | 26 |
| 2.3.4 Vícevrstvá navigační mřížka..... | 26 |
| 2.3.5 Pokročilý systém umělé inteligence..... | 27 |
| 2.3.6 Fyzické přizpůsobení postavy | 27 |
| 2.3.7 Procedurální deformace pohybu a řešení IK..... | 27 |
| 2.4 AUDIO | 27 |
| 2.4.1 Editor ovládání zvuku | 27 |
| 2.4.2 Abstrakce zvuku..... | 28 |
| 2.4.3 HRTF Spatializace zvuku | 29 |
| 2.4.4 Zvukové komponenty..... | 29 |
| 2.4.5 Zvuková okluze | 29 |
| 2.4.6 Dynamický systém odezvy | 29 |

| | | |
|-----------|---|-----------|
| 2.5 | FYZIKA..... | 30 |
| 2.5.1 | Ohýbání vegetace dotykem..... | 30 |
| 2.5.2 | Vestavěný vztlak a simulace vody..... | 30 |
| 2.5.3 | Různé modely destrukcí..... | 30 |
| 2.5.4 | Pokročilá lana..... | 31 |
| 2.6 | VÝKON..... | 31 |
| 2.6.1 | Profilování ve hře..... | 31 |
| 2.6.2 | Správa vláken řízená daty..... | 31 |
| 2.6.3 | Statoskop..... | 31 |
| 2.7 | NOVINKY CRYENGINE 5.7.1..... | 32 |
| 3 | POROVNÁNÍ CRYENGINE..... | 34 |
| 3.1 | POROVNÁNÍ S UNITY..... | 34 |
| 3.1.1 | Výhody a nevýhody..... | 34 |
| 3.2 | POROVNÁNÍ S UNREALENGINE 5..... | 35 |
| 3.2.1 | Výhody a nevýhody..... | 35 |
| II | PRAKTICKÁ ČÁST..... | 37 |
| 4 | TVOBRA DOKUMENTACE V ČESKÉM JAZYCE..... | 38 |
| 4.1 | NÁSTROJE CRYENGINE UVEDENÉ V MANUÁLU..... | 38 |
| 5 | TVORBA APLIKACE..... | 39 |
| 5.1 | NÁVRH APLIKACE..... | 39 |
| 5.2 | TVORBA LEVELU..... | 39 |
| 5.3 | TVORBA TERÉNU..... | 40 |
| 5.4 | VKLÁDÁNÍ ENTIT PROSTŘEDÍ..... | 42 |
| 5.4.1 | Vkládání vegetace pomocí Vegetation Editor..... | 42 |
| 5.4.2 | Vkládání objektů pro terén pomocí nástroje Brush..... | 43 |
| 5.4.3 | Tvorba vodopádu..... | 44 |
| 5.4.4 | Tvorba řeky..... | 46 |
| 5.4.5 | Tvorba cesty..... | 47 |
| 5.4.6 | Vložení objektů prostředí..... | 48 |
| 5.4.7 | Nastavení prostředí levelu a počasí..... | 52 |
| 5.5 | PŘIDÁNÍ AUDIO EFEKTŮ NÁSTROJEM ACE..... | 54 |
| 5.6 | TVORBA CÍLŮ HRY POMOCÍ FLOW GRAPH..... | 56 |
| 5.7 | ÚPRAVA HLAVNÍHO MENU A OBTÍŽNOSTI..... | 61 |
| 5.8 | POUŽITÍ NÁSTROJE LENS FLARE NA OBJEKT SLUNCE..... | 62 |
| 5.9 | VLOŽENÍ A NASTAVENÍ NEPŘÁTELSKÝCH POSTAV..... | 62 |
| 5.10 | FINÁLNÍ ÚPRAVY..... | 64 |
| 5.11 | UŽIVATELSKÝ POHLED NA HRU..... | 66 |
| | ZÁVĚR..... | 67 |
| | SEZNAM POUŽITÉ LITERATURY..... | 68 |
| | SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK..... | 71 |
| | SEZNAM OBRÁZKŮ..... | 72 |
| | SEZNAM TABULEK..... | 74 |
| | SEZNAM PŘÍLOH..... | 75 |

ÚVOD

V dnešní době je herní engine nedílnou součástí zábavního průmyslu. Díky možnostem fyzikálního prostředí, rozpočívání 3D modelů, animací a různých grafických efektů je možno vytvořit nejen moderní a realisticky vypadající hru, ale třeba také CGI efekty pro filmy, seriály či reklamní spoty.

Vývojáři dnes mají k dispozici velké množství herních engine, ze kterých si lze vybrat. A právě díky software a dostatečné hardwarové podpoře je poměrně jednoduché vytvořit funkční počítačovou hru či aplikaci.

Díky neustále se rozvíjejícím počítačovým technologiím lze vytvářet stále realističtější grafické prostředí pro aplikace na počítač, mobil, herní konzole či virtuální realitu. Všechny tyto možnosti CryEngine podporuje.

K počítačové grafice mám poměrně blízko jakožto nadšený hráč počítačových her, kde si užívám kromě samotného příběhu i grafické zpracování. Velký přínos vidím také ve zmiňovaných filmech. Ve své době mě nadchly už tehdejší možnosti CGI použité v seriálu Putování s dinosaury, kde byli dinosauři, které jsem dříve znal jen z uměleckých obrázků odborných knih, zpracování do pohyblivých 3D modelů.

V této bakalářské práci se chci věnovat hernímu engine CryEngine právě z důvodu, abych si tento engine osvojil a naučil se s ním pracovat, a aby má práce případně pomohla dalším studentům či zájemcům o tuto problematiku a usnadnila jim práci s tímto herním engine.

I. TEORETICKÁ ČÁST

1 PŘEDSTAVENÍ A SEZNÁMENÍ S CRYENGINE

CryEngine je herní engine, jehož vývoj započal v roce 1999 v německém městě Koburk vývojářským studiem Crytek. [1] První komerční hra, která byla vytvořena v CryEngine pro společnost Ubisoft, se jmenovala Far Cry. V roce 2007 byla vydána novější verze CryEngine 2, o dva roky později, v roce 2009, společnost Crytek vydala CryEngine 3. Po čtyřleté pauze vyšla v roce 2013 nová verze CryEngine. Tato verze byla natolik odlišná od předchozích verzí, že se společnost Crytek rozhodla vydat tuto verzi bez očíslování. Poslední verze CryEngine V byla vydána v roce 2016 a doposud nebyla nahrazena verzí novou. Tato verze je i předmětem mé bakalářské práce. [2]

Mezi známé herní tituly vydané v CryEngine patří Far Cry, Prey, Kingdom Come: Deliverance, Climb, Hunt: Showdown, Deceit, série Sniper Ghost Warrior a série Crysis.[3]

2 HLAVNÍ VLASTNOSTI A MOŽNOSTI

CryEngine poskytuje velké množství nástrojů pro práci včetně podpory VR. Poradí si dobře s otevřeným světem ve skvělém grafickém zpracování.

Podporuje programovací C++ a C# a skriptovací LUA jazyky. K vytvoření hry není potřeba rozsáhlé znalosti programovacích jazyků, spoustu věcí se dá vytvořit pomocí takzvaných vizuálních skriptů jako Flowgraph a Schematyc. CryEngine má technickou podporu včetně oficiální dokumentace a tutoriálů.

CryEngine je poskytován pro nekomerční účely zdarma. Pro komerční účely se platí autorské poplatky ve výši 5 % z celkového hrubého výdělku. Od autorských poplatků jsou osvobozeny příjmy, které nepřevyšují prvních pět tisíc dolarů za projekt na rok. [4]

2.1 Vizualizace

2.1.1 Oblastní světla

Tento typ světla je explicitně modelován jako plocha se specifickým tvarem, což vede k mnohem větší realističnosti než tradiční bodové zdroje světla. Funkce plošných světel lépe zobrazuje celkové rozložení světla.



Obrázek 1. Oblastní světla [5]

2.1.2 Podpora DirectX 12

CryEngine také podporuje nízkoúrovňové rozhraní API DirectX 12 od společnosti Microsoft. Jedná se o sadu knihoven, které umožňují přímé a efektivní ovládání hardware. Využívá se především pro tvorbu počítačových her a multimediálních aplikací.

2.1.3 Fyzikální vykreslování

Fyzikální vykreslování slouží k simulaci interakcí mezi světlem a materiály pomocí fyziky reálného světa. Simulace chování světla tak jako v reálném světě vede k přirozenějším a věrohodnějším výsledkům a zajišťuje, aby materiály vypadaly věrohodně bez ohledu na světelné podmínky. Výsledkem je reálnější a působivější svět v projektu. Výhodou použití fyzikálně založených modelů je, že k dosažení efektů je zapotřebí méně parametrů, což vede k racionálnějšímu pracovnímu postupu tvorby obsahu.



Obrázek 2. Vykreslování [5]

2.1.4 Dynamická vodní kaustika v reálném čase

Objem vody a řek podporuje jednotný přístup ke kaustice (obálka světelných paprsků odražených nebo zlomených zakřivenou plochou nebo předmětem). Ty zachycují kaustiku 1:1 z vodní hladiny, která v kombinaci s vodními vlnami z objektů interagujících s vodou vytváří věrohodný efekt.



Obrázek 3. Vodní kaustika [5]

2.1.5 Teselace

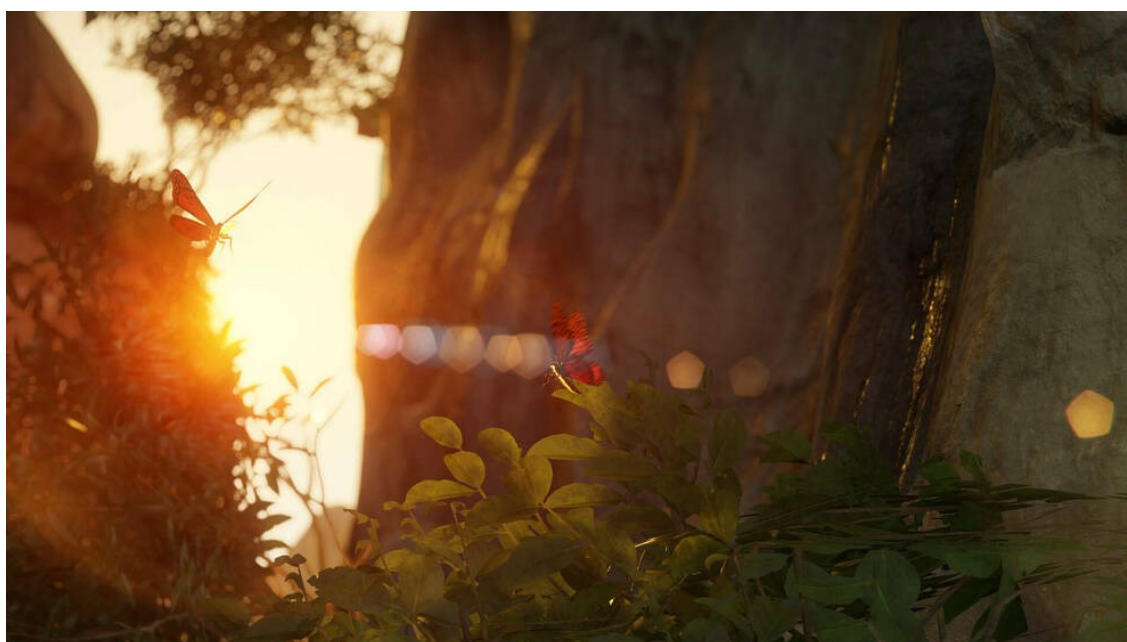
Engine podporuje hardwarově akcelerovanou teselaci na všech objektech, včetně animovaných postav. Podporovány jsou tři různé typy teselace: Phong, PN Triangles a Displacement Mapping. První dvě metody teselace jsou určeny k vylepšování objektů, kdy se vezme hrubý nízkopolymerový model a pomocí teselace se vytvoří realisticky vypadající objekt s vyšší věrností. Třetím a nejvýznamnějším typem teselace je Displacement Mapping. Tato technika nabízí největší kontrolu nad silou a místem, kde dochází k posunutí prostřednictvím výškové mapy. Všechny tyto metody lze v prostředí Sandbox upravovat v reálném čase.



Obrázek 4. Teselace [5]

2.1.6 3D HDR odlesky objektivu

Lens Flare je systém, který jednoduchým a intuitivním způsobem vytváří odlesky podobné těm, které se používají ve filmovém průmyslu. Pomocí vestavěného editoru Flare Editor je možné jediným kliknutím načíst předvolby a přidávat nebo odebírat prvky pro dosažení potřebného vzhledu. Editor je navržen tak, aby poskytoval okamžitou vizuální zpětnou vazbu. Uživatelé mohou kombinovat více předvoleb. Lens flare lze také použít na jakýkoli zdroj světla v úrovni, včetně slunce.



Obrázek 5. HDR odlesky objektivu [5]

2.1.7 Efektivní antialiasing

Engine obsahuje i nejnovější iterace vyhlazování, které minimalizují procházení pixelů a zajišťují stabilní obraz. CryEngine podporuje časový antialiasing (vyhlazení zubatých okrajů), který je dostatečně efektivní pro provoz na všech platformách. Předběžné zpracování map drsnosti zajišťuje minimalizaci zrcadlového aliasingu, zatímco volitelný filtr normálové odchylky v prostoru obrazovky opravuje blikání světla na tenké geometrii. Pro případy, kdy je požadována nejvyšší možná věrnost obrazu, je k dispozici dodatečné vzorkování.



Obrázek 6. Efektivní antialiasing [5]

2.1.8 Rozmazání pohybu a hloubka ostrosti

Motion Blur (rozmazání pohybem) lze aplikovat jak na jednotlivé objekty, tak na celou scénu. Efekty hloubky ostrosti lze snadno definovat v editoru Sandbox. Efekty jsou implementovány v co nejvyšší kvalitě při nízkých nákladech na výkon na všech podporovaných platformách.



Obrázek 7. Rozmazání pohybu [5]

2.1.9 Místní reflexe v reálném čase

Odrazy jsou jedním z největších problémů při vykreslování v reálném čase, zejména u engineů s odloženým vykreslováním (výpočet osvětlení se provádí až po výpočtu geometrie). CryEngine obsahuje vylepšenou technologii odrazů, takzvaný Screen Space Reflections (SSR). SSR odhaduje paprskové HDR odrazy pro místní objekty, což umožňuje, aby jakýkoli druh povrchu ve scéně efektivně odrazil své blízké okolí v reálném čase.



Obrázek 8. Místní reflexe [5]

2.1.10 Realistická vegetace

Vylepšený systém vegetace umožňuje vytvořit velké množství stromů, keřů a rostlin, které přesně odrážejí fyzikální síly ve světě. Tato funkce vnáší život do obvykle statických scén. Dokáže simulovat jednotlivá stébla trávy a interagovat s hráčem, umělou inteligencí, dynamickými objekty, výbuchy a větrem.



Obrázek 9. Vegetace [5]

2.1.11 Globální osvětlení založené na voxelech

Jedná se o systém globálního osvětlení založený na voxelech. Rozšiřuje stávající osvětlení založené na obrazu (IBL) o rozsáhlou okluzi (vystavení osvětlení) okolí a nepřímý odraz světla v reálném čase. Oproti tradičním předpřipraveným řešením nevyžaduje Sparse Voxel-Based Global Illumination (SVOGI) žádné dlouhé předběžné výpočty ani ukládání na disk. To znamená, že jej lze použít pro obrovská prostředí s otevřeným světem a dynamickou denní dobou, a zároveň zjednodušuje uživatelům návrh úrovní tím, že jim umožňuje okamžitě testovat osvětlení v reálném čase.

2.1.12 Mapy stínů pro jednotlivé objekty

Mapy stínů umožňují vytvořit vlastní mapy, které lze přiřadit libovolné entitě v úrovni. To může vést ke zvýšení kvality stínů díky vyšší hustotě textur stínů ve světě vytvořeného levelu a menšímu rozsahu hloubky. Je to užitečné zejména pro cutscény (herní události) nebo zahrnutí dodatečné definice stínů u konkrétních objektů. Tato funkce je dostupná prostřednictvím speciálního uzlu Flowgraph, který návrhářům poskytuje snadný přístup k dynamickým stínům v reálném čase.



Obrázek 10. Mapy stínů [5]

2.1.13 Směrová okluze v prostoru obrazovky

Jedná se o technologii Screen Space Directional Occlusion (SSDO) ve hrách v reálném čase. Nyní SSDO umožňuje zahrnout do výpočtu Ambient Occlusion (realistické dopadání světelných paprsků) libovolný počet světel a zohlednit barvu všech světel. SSDO zlepšuje přesnost fyzického osvětlení a v konečném důsledku umožňuje vrhat stíny z každého jednotlivého zdroje světla v prostředí.



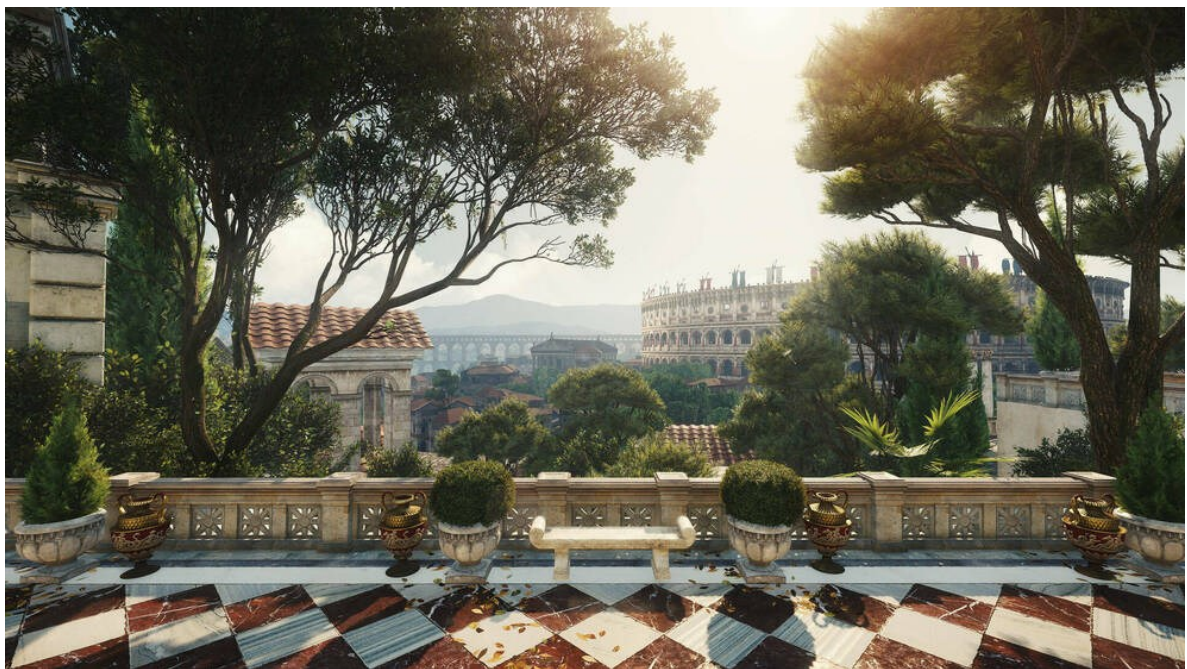
Obrázek 11. Směrová okluze [5]

2.1.14 Filmové mapování tónů HDR

Nejnovější verze mapovače tónů zavádí uživatelem řízenou křivku HDR, která umožňuje lepší napodobování chování filmové reprodukce barevných tónů. Nová nastavení umožňuje uživatelům přesně nastavit světlé a tmavé hodnoty. Tato funkce umožňuje širší tonální rozsah s možností zlepšit definici tmavých, světlých a středních tónů přímým způsobem a poskytuje uživateli lepší kontrolu nad expozicí scény.

2.1.15 Osvětlení na základě obrazu

Osvětlení na základě obrazu (IBL) přináší lepší osvětlení a vykreslování všech materiálů. Stínovací model CryEngine využívá pokročilé hodnoty funkce obousměrného rozložení odrazu a normalizaci zrcadlových světel k vykreslení velmi složitých světelných situací. Na rozdíl od tradičtějšího nastavení využívá IBL zrcadlovou odrazivost a drsnost materiálů a vrací tak fyzikálně správnější vzhled. Tvrdé zrcadlové odrazy se stávají stejně jednoduchými jako jediné vyhledání textury, zatímco rozmazané odrazy lze efektivně aproximovat úpravou úrovně mipmap během vyhledávání.



Obrázek 12. Osvětlení na základě obrazu [5]

2.1.16 Systém částicového efektu

Systém částic je v nejnovější verzi CryEngine navržen tak, že volitelně přenáší správu a zpracování částic na grafický procesor. Zároveň umožňuje uživatelům dosáhnout výrazně vyšší propustnosti částic pro nákladné efekty, což zvyšuje výkon.



Obrázek 13. Systém částicového efektu [5]

2.1.17 Objemové stíny mlhy

V CryEngine je také podporována technika globální mlhy a stínů mraků, která výrazně zlepšuje atmosférické podmínky v mlhavém prostředí. Tato funkce je mimořádně výkonná, protože vykresluje stíny objektů a mraků do samotné objemové mlhy.



Obrázek 14. Objemové stíny mlhy [5]

2.1.18 Mapování paralaxní okluze

Tato technika, která dodává materiálům větší prostorovou definici, patří mezi další vlastnosti CryEngine. Parallax Occlusion Mapping (POM) a Offset Bump Mapping (OBM), které jsou podobné stávajícím technikám Tessellation a Displacement, ale jsou mnohem méně náročnější na výkon. S pokročilou možností Silhouette Parallax Occlusion Mapping je možné vytvořit efekt 3D povrchu, ale také ovlivnit siluetu neboli vykreslenou geometrii povrchu, na který je aplikována. [5]

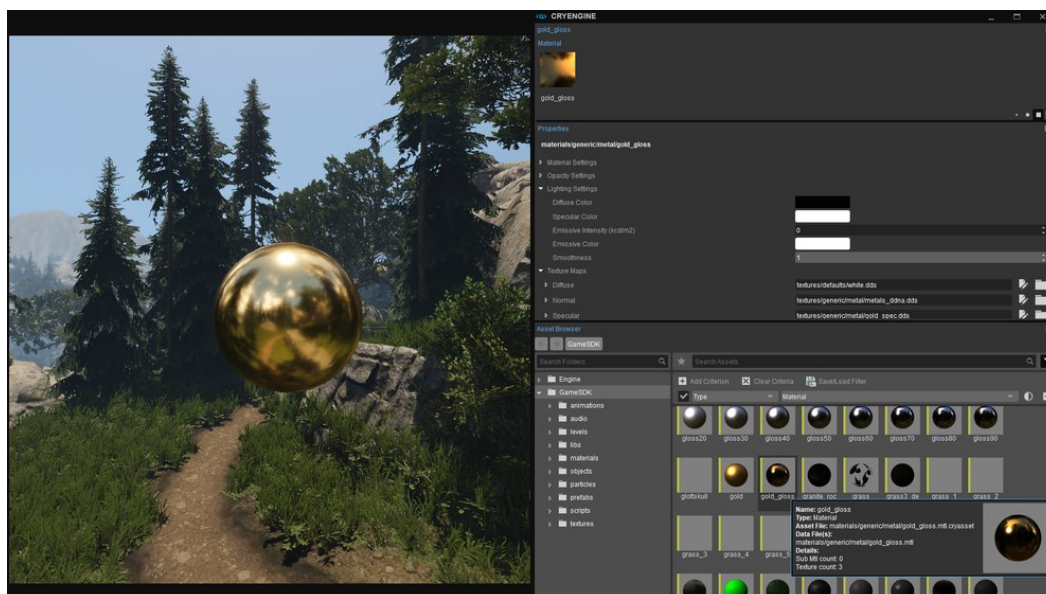
2.2 Sandbox

2.2.1 CryEngine Sandbox

Sandbox umožňuje uživatelům kontrolu nad jejich projekty v reálném čase a nabízí řadu efektivních nástrojů. Sandbox lze rozšířit pomocí skriptů a zásuvných modulů tak, aby vyhovovaly specifickým potřebám uživatele.

2.2.2 Editor materiálů

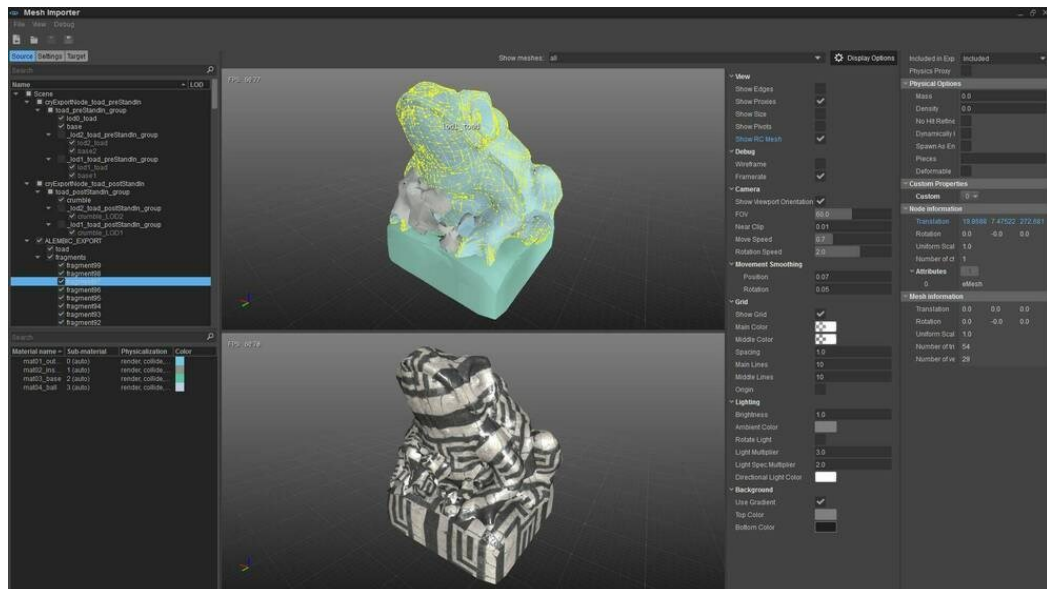
V tomto editoru je možné nalézt velké množství optimalizovaných shaderů, stejně tak je možné přidávání vlastních shaderů pro větší výkon a flexibilitu.



Obrázek 15. Editor materiálů [6]

2.2.3 Podpora FBX

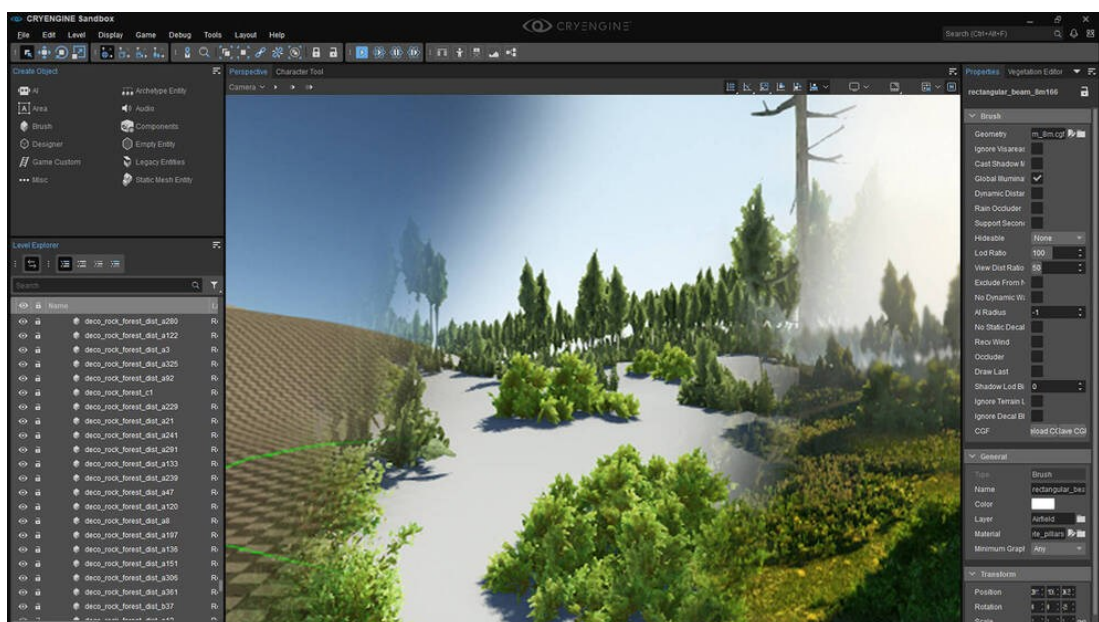
CryEngine podporuje import souborů z jakéhokoli nástroje pro tvorbu digitálního obsahu pomocí preferovaného formátu FBX.



Obrázek 16. Podpora FBX [6]

2.2.4 Intuitivní design úrovní

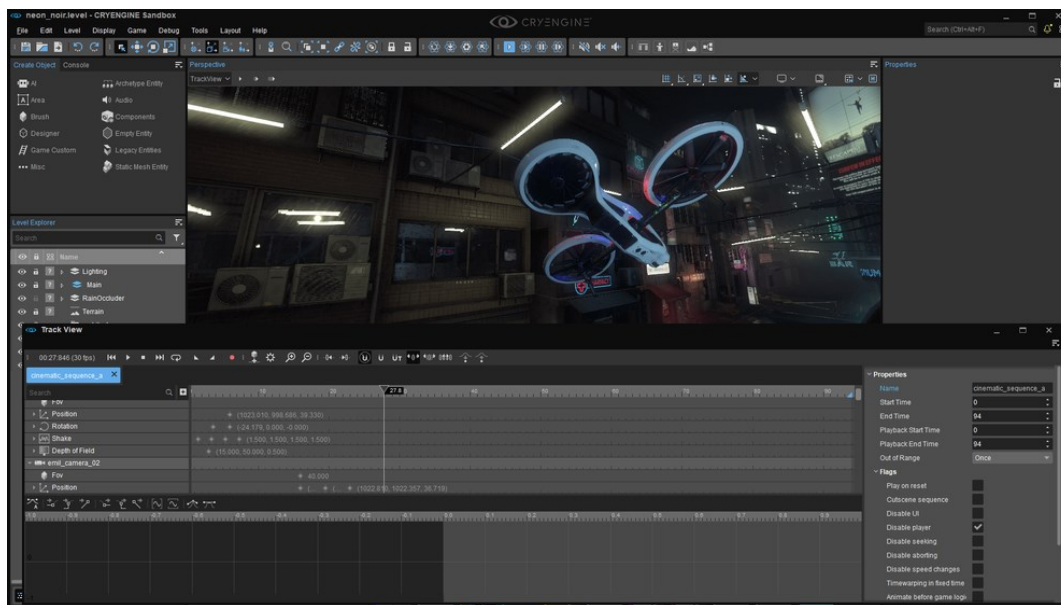
CryEngine poskytuje výkonné a intuitivní nástroje pro tvorbu úrovní podle představ uživatele. Prototypy je možné vytvářet pomocí nástroje Designer pro modelování v editoru nebo lze vyplnit úroveň procedurálně pomocí nástroje Vegetace.



Obrázek 17. Design úrovní [6]

2.2.5 Filmový editor Trackview

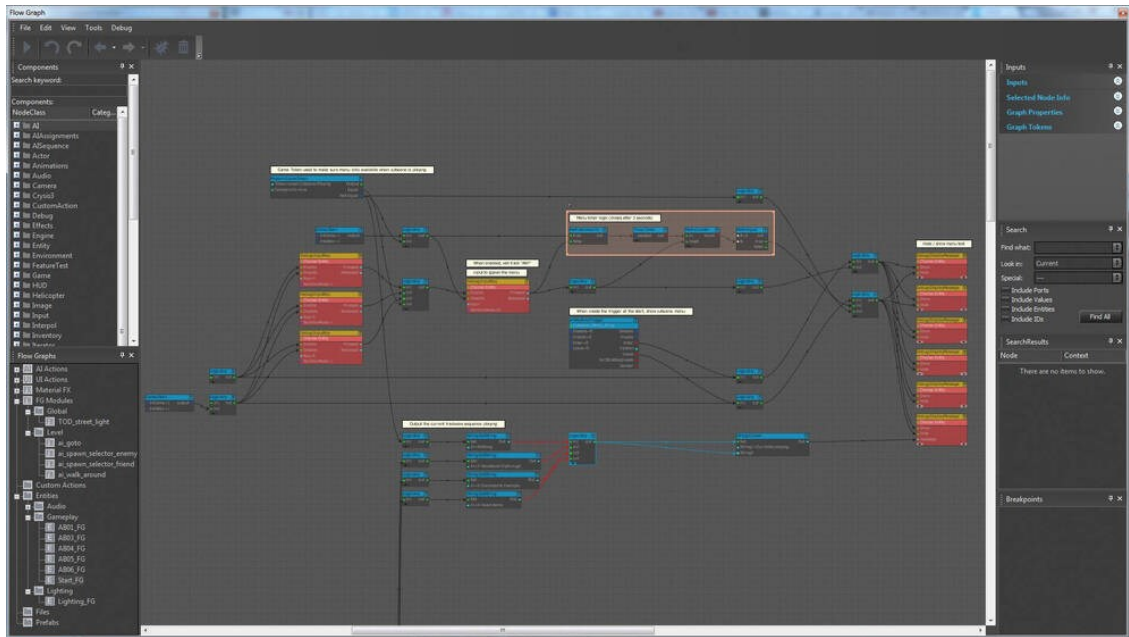
Editor TrackView je vestavěný nástroj pro střih scén, který slouží k vytváření interaktivních filmových sekvencí s časově závislým ovládním objektů a událostí. Vytváření filmových nebo skriptovaných událostí je snadné a umožňuje uživateli sekvencovat objekty, animace a zvuky ve scéně.



Obrázek 18. Trackview [6]

2.2.6 Flowgraph

Jedná se o jednoduchý, ale výkonný vizuální skriptovací systém CryEngine. Poskytuje uživateli intuitivní rozhraní pro vytváření a ovládní herní logiky a událostí. S přidáním vylepšení pracovního postupu, kterými jsou ladění v reálném čase, body přerušování a moduly, se uživatel může soustředit na iteraci a zlepšování hratelnosti.



Obrázek 19. Flowgraph [6]

2.2.7 Integrace látek

Integrace látkových materiálů CryEngine umožňuje importovat soubory přímo do enginu, což poskytuje flexibilitu a možnost ovládat způsob generování materiálů v samotném enginu. [6]



Obrázek 20. Integrace látek [6]

2.3 Umělá inteligence a animace

2.3.1 Technologie postavy

CryEngine poskytuje škálovatelnější, technicky pokročilejší animační a vykreslovací systémy, které umožňují vytvoření realistické postavy pro hry v reálném čase. Mezi tyto pokročilé funkce stínování patří: podpovrchové rozptylování v prostoru obrazovky, realistické paralaxní mapování duhovky pro zobrazení očí a sjednocené stínovače pro stíny postav a vlasů.

2.3.2 Parametrická skeletová animace

Díky míchání různých výchozích pohybů na základě uživatelem definovaných parametrů poskytuje CryEngine citlivé, interaktivní a detailní ovládání kosterních animací. Postavy se pohybují v prostředí a reagují za běhu na změny pohybu, směru nebo rychlosti prolínáním animací, čímž vytvářejí realistickou reprezentaci pohybu.

2.3.3 Geometrická mezipaměť

Jedná se o podporu ukládání a přehrávání složité simulované geometrie pomocí techniky ukládání bodů do mezipaměti, která byla poprvé vyvinuta pro animované filmy. Tento přístup obvykle vede k datům nezvládnutelné velikosti, ale technologie CryEngine ji snižuje na 10 % původní velikosti souboru. Interpretovaná "GeomCache" vykresluje stejně efektivně jako statická geometrie, přičemž je možná komprese Lossy-to-Lossless, podpora instancování, streamování a přehrávání ve vyrovnávací paměti.

2.3.4 Vícevrstvá navigační mřížka

Vícevrstvá navigační síť (MNM) je vylepšená dynamická navigační datová struktura, kterou umělá inteligence používá k hledání cesty v úrovních. Tento systém je integrován do systému CryEngine Sandbox, což znamená, že uživatel definuje oblast, kde má být navigační síť vygenerována, a zbytek řídí MNM. MNM také podporuje dynamické aktualizace ve hře v reálném čase, takže když se oblast změní, navigační síť se podle toho bez prodlení aktualizuje.

2.3.5 Pokročilý systém umělé inteligence

CryEngine obsahuje výkonnou technologii AI, která zvládá chování postav s modulárními smyslovými systémy, jako je sluch a zrak, a podporuje požadavky pohybového systému. Umělá inteligence nevyžaduje další middleware a je plně integrována.

2.3.6 Fyzické přizpůsobení postavy

Simulační algoritmy byly přizpůsobeny pro efektivní, stabilní a rychlé zpracování postav tak, aby nabízely vysoký stupeň realismu při vysokých snímkových frekvencích. Nové flexibilní kolize umožní lépe sladit modely a nabídnout realistickou interakci s postavami při zachování vysokého výkonu.

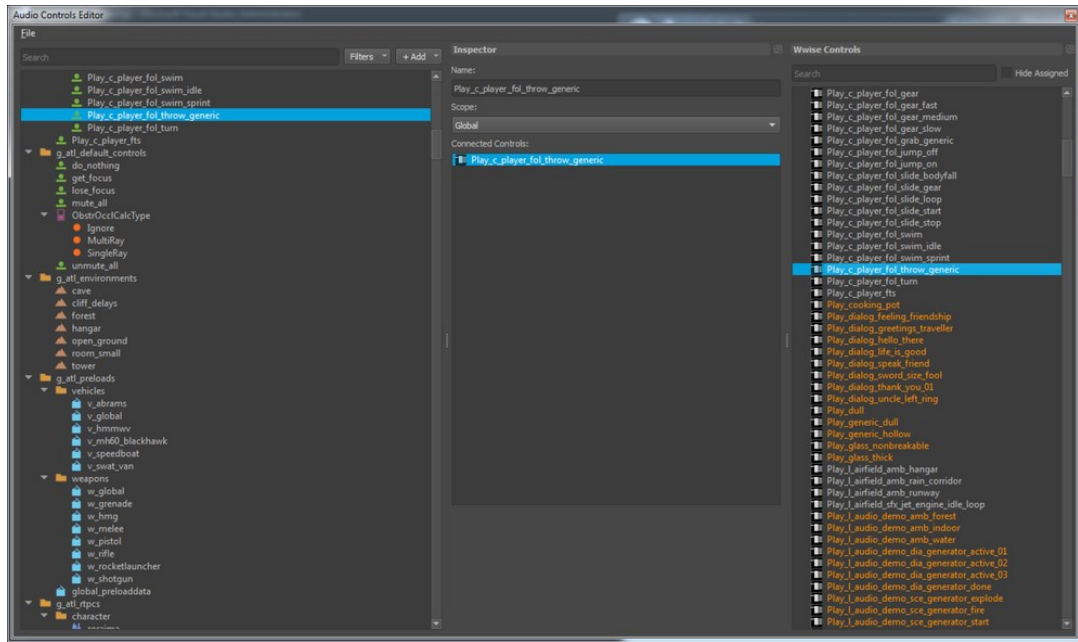
2.3.7 Procedurální deformace pohybu a řešení IK

Procedurální algoritmy, jako je CCD IK, analytická IK, IK založená na příkladech nebo fyzikální simulace, se používají k doplnění předem vytvořených animací, aby se zabránilo typickému počítačovému vzhledu. Škálovatelná technika warpingu zachovává styl a obsah základního pohybu a zároveň plně zohledňuje omezení daná interaktivním prostředím. [7]

2.4 Audio

2.4.1 Editor ovládání zvuku

CryEngine Sandbox obsahuje Editor ovládání zvuku (ACE). Tento výkonný nástroj umožňuje rychlé flexibilní vytváření a připojování ve stylu drag and drop: událostí, přepínačů, stavů a RTPC, jakož i správu nahraných zvukových záznamů. Změny v ACE jsou aktualizovány v reálném čase, zvuk se okamžitě přehrává a není nutné restartovat aplikaci. ACE má plně funkční komponentu pro vyhledávání a filtrování, která umožňuje rychle a jednoduše najít, zobrazit nebo upravit i obrovské množství zvukových událostí.



Obrázek 21. ACE [8]

2.4.2 Abstrakce zvuku

CryEngine také podporuje komunikaci se zvukovým middlewarem, například CRI ADX2, FMOD Studio, Miles Sound System nebo Wwise. CryEngine zároveň obsahuje bezplatnou alternativu v podobě implementace SDL_mixer. Jmenované middlewarové nástroje jsou podporovány standardně a lze je jednoduše aktivovat.



Obrázek 22. Abstrakce zvuku [8]

2.4.3 HRTF Spatializace zvuku

HRTF poskytuje plné využití 3D zpracování zvuku v reálném čase, které poskytuje přesné akustické vlastnosti šíření zvuku v reálném světě.

2.4.4 Zvukové komponenty

Pomocí zvukových komponent mohou uživatelé využívat plný výkon zvuku včetně míchání dozvuků a atmosféry na základě oblasti, bodových FX a manipulace s ovládáním zvuku.

2.4.5 Zvuková okluze

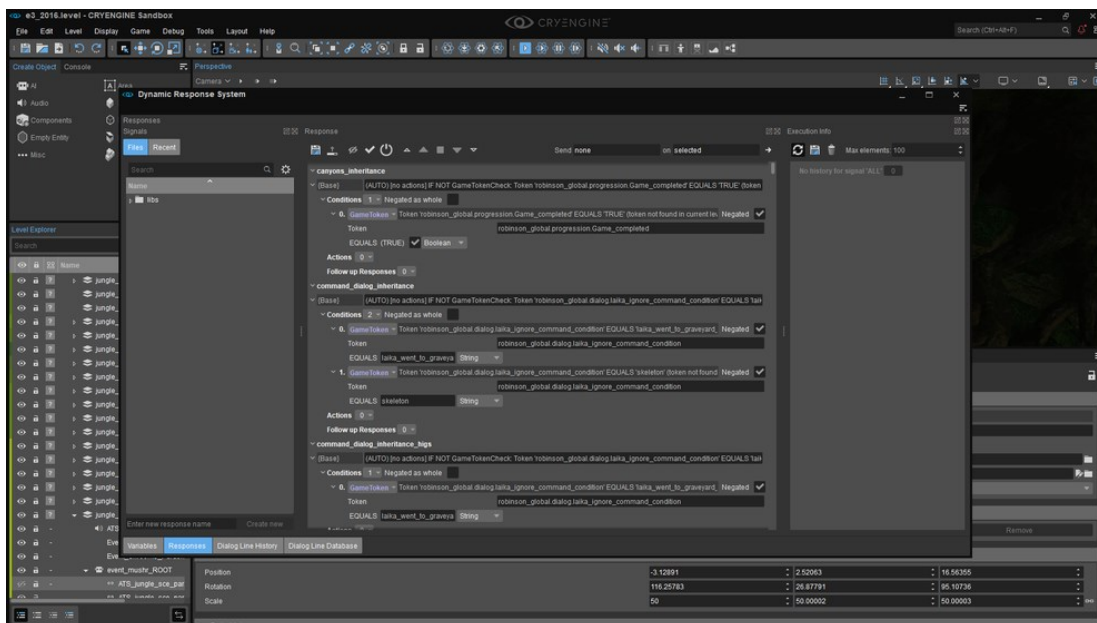
V engine je také integrovaná podpora přímé okluze zvuku. V souvislosti s tím je umožněno využívat podporovaný zvukový middleware jako například CRI ADX2, FMOD Studio, Miles Sound System nebo Wwise.



Obrázek 23. Příklad zvukové okluze [8]

2.4.6 Dynamický systém odezvy

Systém dynamické odezvy (DRS) představuje způsob implementace dialogů ve hře. Odděluje vztah dialogu a zpětné vazby od herní logiky. DRS toho dosahuje přidáním abstraktní vrstvy mezi něco, co se děje, a probíhající zpětnou vazbu dialogu. DRS lze také použít k využití jakéhokoli jiného systému, jako jsou například animace, částice nebo fyzika. [8]



Obrázek 24. Dynamický systém odezvy [8]

2.5 Fyzika

CryEngine obsahuje kompletní fyzikální řešení, od pevných entit až po objekty a destrukci, které umožňují oživit úroveň a hratelnost. Řešení poskytuje podporu pro širokou škálu funkcí, včetně libovolných sítí na dynamických objektech a speciálního režimu simulace pro kloubové struktury.

2.5.1 Ohýbání vegetace dotykem

Funkce Vegetation Touch Bending umožňuje jednotlivým prvkům vegetace fyzickou interakci s objekty, hráči a větrem.

2.5.2 Vestavěný vztlak a simulace vody

CryEngine poskytuje přesné výpočty vztlaku pro vodu, vzduch a libovolné tvary objektů. Speciální podpora je poskytována pro vlastní objemy vody, řeky a oceánské vlny. Engine také podporuje simulaci vody, která se dokáže automaticky přizpůsobit základní geometrii plavidla.

2.5.3 Různé modely destrukcí

CryEngine poskytuje různé modely realisticky vypadajících destrukcí. To je umožněno kombinací předem určených zlomů spojených pomocí tahových spojů a přímou úpravou materiálu pomocí dynamického Booleova odečítání.

2.5.4 Pokročilá lana

Nástroj Rope Tool podporuje lana s různými režimy kolize a upevnění. Dokáže automaticky sestavovat lana z opakujících se segmentů.[9]

2.6 Výkon

2.6.1 Profilování ve hře

Profilování ve hře umožňuje okamžité zjištění, kde a kdy se objevily problémy s výkonem. Jednotlivé informace se řídí hierarchií, která okamžitě odhalí, o jaké moduly a funkce se jedná. Další informace o době trvání a výskytu pomáhají vyhodnotit jejich dopad.



Obrázek 25. Profilování [10]

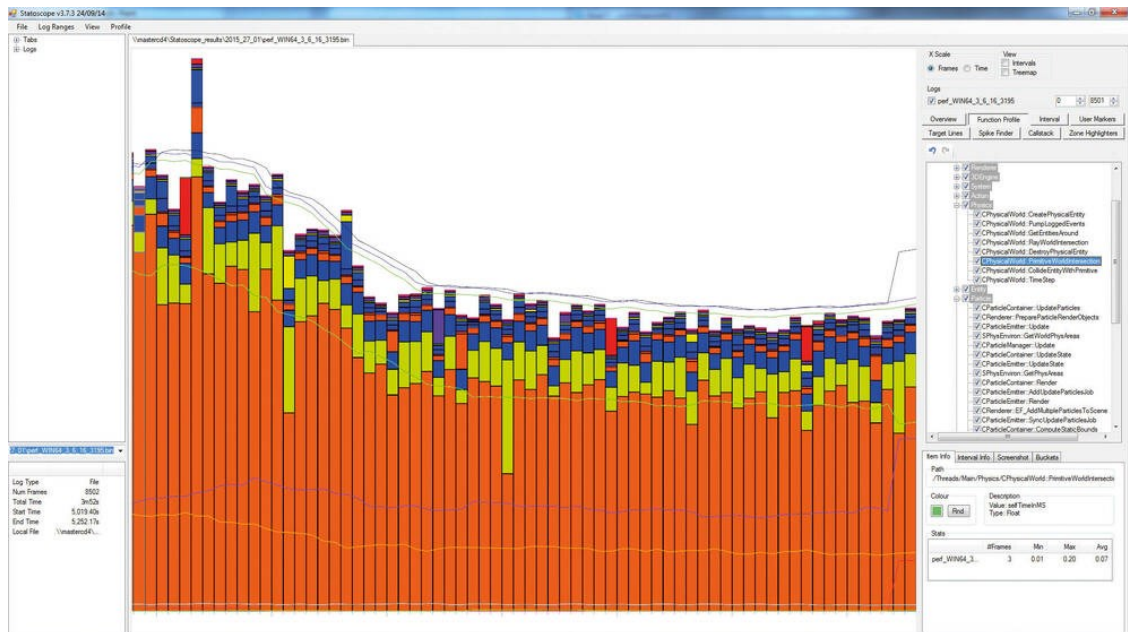
2.6.2 Správa vláken řízená daty

CryEngine se stará o efektivní rozdělení práce mezi softwarová a hardwarová vlákna. Jedno centrální místo pro vytváření vláken umožňuje plánovat a sledovat paralelní výpočty za běhu. Využití vláken je nezbytné pro aplikace reálného času, aby bylo možno plně využít vícejadrový hardware.

2.6.3 Stoskop

Výkonný interní diagnostický nástroj Statoscope slouží k získání informací o tom, kde se spotřebovává výkon. Přidáním dat do nástroje Statoscope může uživatel zobrazit funkce zobrazení statistik na základě jednotlivých snímků v externím nástroji. Buď může data

shromažďovat přímo z enginu do nástroje Statoscope, nebo může informace zaznamenat do souboru a přečíst je později. Uživatelé navíc mohou překrýt dvě relace snímání na sebe a spustit porovnání všech změn provedených mezi oběma relacemi. [10]



Obrázek 26. Statoscope [10]

2.7 Novinky CryEngine 5.7.1

Ve verzi CryEngine 5.7.1 byla integrována nová Scaleform 4, která je vylepšením Scaleform 3. Tato nová verze nabízí lepší výkon vykreslování pro všechny platformy. Je zpětně kompatibilní s jazykem ActionScript 2.0 a zároveň podporuje jazyk ActionScript 3.0, který zase umožňuje používat Adobe Animate. Scaleform 3 je ovšem stále k dispozici pro případ chyb či nefunkčnosti Scaleform 4 v projektu.

Scaleform Schematyc byl přidán pro účely emulace nezbytných funkcí, které poskytuje Flow Graph pro systém uživatelského rozhraní. Prvky uživatelského rozhraní jsou k dispozici jako komponenty, zatímco funkce a události odpovídající těmto prvkům lze definovat v souboru Element XML. Podporuje Scaleform 3 a 4.

Rovněž byl aktualizován modul GamePlatform, který byl dodáván se systémem CryEngine 5.6. Aktualizované funkce zahrnují podporu různých platforem, jako jsou Steam, PlayStation a Xbox. Nově obsahuje volitelný zásuvný modul, který poskytuje vizuální skriptovací uzly pro Flow Graph a Schematyc. Discord Rich Presence SDK byl aktualizován na Discord Game SDK.

Kód enginu a CMake byly aktualizovány tak, aby podporovaly Visual Studio verzí 2017, 2019 a 2022, zatímco podpora Visual Studio 2015 byla vyřazena. Spolu s touto aktualizací byla přidána podpora implementace standardu C++17, zatímco sada Windows SDK byla aktualizována na verzi 10.0.20348.0. [11]

3 POROVNÁNÍ CRYENGINE

3.1 Porovnání s Unity

Unity je software používaný především k vývoji videoher pro počítačové a mobilní platformy. Používá se také v oblasti rozšířené a virtuální reality. Unity bylo vyvinuto společností Unity Technologies v roce 2005. Jedná se o multiplatformní herní engine. Původně byl navržen pouze pro operační systém OS X, ale nyní jej lze používat na různých platformách, jako jsou Windows, Mac atd. Unity je zdarma a je přístupný všem. Snadno se používá, takže jej využívá většina herních vývojářů pro tvorbu her a aplikací. Je napsán v jazycích C++ a C#. Používá se k vytváření 2D a 3D her.

3.1.1 Výhody a nevýhody

Výhody Unity:

- Jedná se o multiplatformní engine pro vývoj her, což znamená, že hry lze používat na více platformách.
- Je nejlepší pro nezávislé vývojáře her, protože jeho osobní licence je zdarma.
- K dispozici je velké množství zásuvných modulů, které rozšiřují jeho možnosti.
- Má vlastní obchod, ze kterého lze stahovat předpřipravené herní produkty.

Nevýhody Unity:

- Vykreslování v Unity je ve srovnání s CryEngine pomalejší.
- Import šablon z jiných platform není povolen, takže je třeba vše vytvořit od začátku.

Výhody CryEngine:

- Snadno se používá a vyžaduje pouze základní znalosti programování.
- Podporuje funkce VR.
- Umožňuje integraci s mnoha softwary, jako je Blender, Maya atd.
- Je vhodný pro hry AAA.

Nevýhody CryEngine:

- K plynulému běhu softwaru je zapotřebí výkonný procesor.

Vyžaduje grafickou kartu s nejnovějším ovladačem. [12]

Tabulka 1 Unity vs CryEngine [12]

| Unity | CryEngine |
|---|--|
| Společnost Unity poskytuje osobní licenci zdarma a profesionální licenci za 1800 dolarů ročně, tj. 150 dolarů měsíčně | Je k dispozici za 9,90 dolarů za měsíční předplatné |
| Podíl na trhu činí 11 % | Podíl na trhu činí přibližně 5 % |
| Požívá programovací jazyky C# a C++ | Požívá programovací jazyky C# a C++ a LUA |
| Nemá kód s otevřeným zdrojovým kódem | Má kód s otevřeným zdrojovým kódem, který je k dispozici na serveru GitHub |
| Má velkou komunitu, která čítá přibližně 210 tisíc členů | Ve srovnání s Unity má menší komunitu |
| K dispozici pro Windows i Mac | K dispozici pouze pro operační systémy Windows |

3.2 Porovnání s UnrealEngine 5

Unreal Engine je počítačový software vytvořený za účelem vývoje her, jedná se tedy o herní engine. Zásahu na vývoji tohoto softwaru má společnost Epic Games a poprvé byl uveden na trh v roce 1998. Jeho autorem je Tim Sweeney. Je schopen vytvářet 2D a 3D hry. Je k dispozici pro různé operační systémy, jako je Windows, MacOS atd. Snadno se používá a je schopen vytvářet komplexní hry.

3.2.1 Výhody a nevýhody

Výhody Unreal Engine:

- Má snadné rozhraní, s nímž lze snadno začít vyvíjet hry.
- Obsahuje implementovaný uzlový editor, který umožňuje vývoj hry bez psaní kódu.
- Dokáže poskytovat velmi kvalitní grafické výstupy v reálném čase. Má lepší technologii vykreslování, která je poměrně rychlejší než ostatní herní enginy.

- Dodává se s předpřipravenými herními úrovněmi, které vývojářům her poskytují základní kostru aplikace.

Nevýhody Unreal Engine:

- Účtuje si 5% licenční poplatek, jakmile uživatelem publikovaná hra přesáhne výdělek 1 milion dolarů.
- Poskytuje méně zásuvných modulů než Unity.
- Potřebuje výkonný procesor spolu s vyhrazenou grafickou kartou a pamětí.

Výhody CryEngine:

- V současné době je CryEngine nejpoužívanějším nástrojem pro tvorbu počítačových her.
- Snadno se používá a vyžaduje pouze základní znalosti programování.
- Podporuje funkce VR.
- Umožňuje integraci s mnoha softwary, jako je Blender, Maya atd.
- Je vhodný pro hry AAA.

Nevýhody CryEngine:

- K plynulému běhu softwaru je zapotřebí výkonný procesor.
- Vyžaduje grafickou kartu s nejnovějším ovladačem. [13]

Tabulka 2 Unreal Engine vs CryEngine [13]

| Unreal Engine | CryEngine |
|--|---|
| Používání je zdarma, ale při překročení výdělku na hře 1 milion dolarů je účtován 5% licenční poplatek | Je k dispozici za 9,90 dolarů za měsíční předplatné |
| Podíl na trhu činí 23 % | Podíl na trhu činí přibližně 5 % |
| Požívá programovací jazyky C++ a Blueprint | Požívá programovací jazyky C# a C++ a LUA |
| 2. místo v herním průmyslu. | 7. místo v herním průmyslu. |
| K dispozici pro Windows i Mac | K dispozici pouze pro operační systémy Windows |

II. PRAKTICKÁ ČÁST

4 TVOBRA DOKUMENTACE V ČESKÉM JAZYCE

Dokumentace vznikla jako součást bakalářské práce. Tato dokumentace byla vytvořena za pomoci stránek oficiální dokumentace CryEngine od společnosti CRYTEK, které sloužily jako vzor. [14] Z oficiální dokumentace byly vybrány ty nejpodstatnější nástroje, instalace CryEngine 5.7.1 LTS pomocí CryEngine Launcher verze 2.1.3 a tyto informace byly doplněny o poznatky na základě zkušenosti práce s engine. Dokumentace odkazuje na internetové stránky, ze kterých bylo čerpáno, včetně odkazů na jednotlivé kapitoly pro podrobnější informace k dané problematice.

4.1 Nástroje CryEngine uvedené v manuálu

CryEngine obsahuje velké množství nástrojů pro práci a editaci levelu v engine. V manuálu jsou popsány následující nástroje:

- Create Object,
- Flow Graph,
- Lens Flare Editor,
- Material Editor,
- Terrain Editor,
- Vegetation Editor.

Tyto nástroje budou použity a demonstrovány v praktické části této bakalářské práce.

CRYENGINE 5.7.1 LTS DOKUMENTACE



5 TVORBA APLIKACE

K tvorbě aplikace a jejího levelu v engine je použit **CryEngine GameSDK Sample Project**, který je k dispozici ke stažení v databázi prostředků. [15] Tento ukázkový projekt je zvolen z důvodu, že oproti standardním šablonám, které jsou k dispozici ihned po instalaci engine, nabízí velké množství objektů, materiálů, modelů postav a vegetace. Ukázkový projekt je určen pro začínající uživatele CryEngine tak, aby tvorba prvních projektů byla uživatelsky co nejjednodušší. Nabízí také rozumně vypadající HUD a základní hlavní menu, které lze použít a editovat podle potřeb uživatele CryEngine.


5.1 Návrh aplikace

Aplikace, která je součástí této bakalářské práce, bude vytvořena jako FPS (First person shooter) s použitím co největšího množství efektů a nástrojů tak, aby byla demonstrována praktická práce s těmito nástroji k dosažení požadovaného autorova výsledku.

Vzhledem k tomu, že aplikace je definována jako FPS, bude pohled hráče z pohledu první osoby. Cílem hry bude dosažení kontrolních bodů a interakce s těmito body, které budou definovány jako mise hry. Hra tedy nabídne hráči dva možné způsoby splnění dané mise, buď přímým postupem a eliminací nepřátelských postav, anebo opatrnějším postupem po mapě, při kterém si bude muset hráč dát pozor na detekování nepřítelem. Hra má celkem čtyři mise.

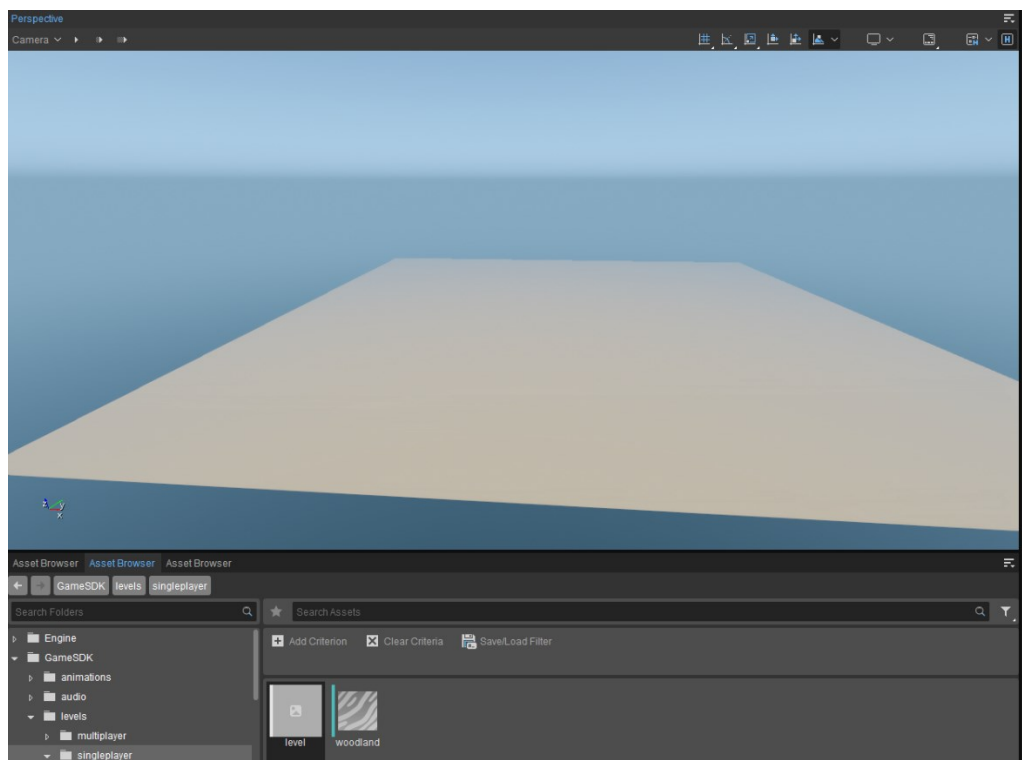
5.2 Tvorba levelu

Tato kapitola byla tvořena s pomocí knihy CryEngine 1.3 Cookbook [16], kniha je věnována CryEngine 3, tudíž některé informace nebyly aktuální, a tak byly dodatečné informace čerpány z online tutoriálů [17] [18].

Po stažení prostředku **CryEngine GameSDK Sample Project** je potřeba vytvořit projekt. Projekt lze vytvořit v menu **Launcheru** v záložce **Asset library**. Po vyhledání tohoto prostředku v knihovně je potřeba kliknout na ikonu  a poté z kontextové nabídky vybrat možnost **Generate project from asset**. Po zobrazení okna **Crate project** je potřeba vyplnit název projektu a cestu v adresáři, kde bude projekt uložen.

Po načtení Sandbox Editoru v okně **Asset browser** ve složce **GameSDK/levels** byly odstraněny ukázkové levely (složka *_testmaps*, *multiplayer/airfield* a *singleplayer/woodland*).

Nový level byl vytvořen pomocí nástroje **File** → **New** → **Level...** Po zobrazení okna **Create level** bylo vybráno umístění ve složce levels a název levelu byl nastaven na level. Po zobrazení nového okna **New Level** nastaveny parametry **Heightmap resolution** na 512x512 a **Texture dimensions** na 1024x1024. Poté dojde k vygenerování nového levelu s rovnou plochou bez textur, kterou je potřeba upravit. V levém okně **Level Settings** byl nastaven parametr **Detail Layers View** na hodnotu 5 pro nastavení vzdálenosti vykreslování textur jednotlivých vrstev.





Obrázek 28. Nový level

5.3 Tvorba terénu

Vygenerované ploše byla přidělena základní textura pomocí nástroje **Tools** → **Terrain Editor** → **Paint**. Ve vlastnostech vrstvy Grass byly změněny následující parametry:

- **Texture** – Vybrána textura *GameSDK/textures/terrain/detail/grass*,
- **Material** – Vybrán materiál *GameSDK/materials/terrain/grass_1*.

Vrstva Grass po spuštění v herním módu působí vybledlým dojmem bez barvy, to lze upravit v parametru materiál kliknutím na ikonu  **Material** *grass_1.mtl*  a následně provést nastavení pro materiál s následujícími parametry:

- **Difuse color** - "r": 12, "g": 147, "b": 66, "a": 255,
- **Emissive color** - "r": 33, "g": 148, "b": 81, "a": 255.

Následně bylo provedeno vyvýšení terénu pomocí nástroje **Tools** → **Terrain Editor** → **Raise/Lower** s nastavením parametrů **Radius (outside)** na 26, **Hardness** na 1 a **Height** na 9 tak, aby okraje vygenerované plochy tvořily přirozené ohraničení herní plochy a celkové nerovnosti terénu tak, aby výsledný terén tvořil různotvárné prostředí. Při nastavení parametru **Height** na zápornou hodnotu dojde ke snižování výšky terénu.

V nástroji **Tools** → **Terrain Editor** → **Paint** byla vytvořena další vrstva s názvem **Rock**, která je použita pro nastavení textury ohraničení mapy horami s následujícími parametry:

- **Texture** – Vybrána textura *GameSDK/textures/terrain/detail/textures/terrain/detail/rock_3d*,
- **Material** – Vybrán materiál *GameSDK/materials/terrain/materials/terrain/rock_4*.

Stejně jako u vrstvy **Grass** bylo provedeno u vrstvy **Rock** nastavení pro materiál s následujícími parametry:

- **Difuse color** - "r": 99, "g": 99, "b": 99, "a": 255,
- **Emissive color** - "r": 72, "g": 72, "b": 72, "a": 255.

Pomocí nástrojů **Tools** → **Terrain Editor** → **Sculpt** → **Flatten** a **Tools** → **Terrain Editor** → **Sculpt** → **Smooth** bylo provedeno vyrovnaní a zhlazení terénu k výsledné podobě jako na obrázku 29.



Obrázek 29. Pohled na upravený level z hráčovy perspektivy

5.4 Vkládání entit prostředí

Aby herní mapa nepůsobila prázdným dojmem, je potřeba do levelu vložit různé entity prostředí. Do levelu budou vloženy budovy, vegetace a entity objektů jako třeba kameny. Pro tyto účely bude použito nástrojů **Create Object** → **Brush** pro vkládání jednotlivých entit a **Tools** → **Vegetation Editor**, který je vhodný zejména pro vkládání více entit najednou na definované ploše rádiusem.

5.4.1 Vkládání vegetace pomocí Vegetation Editor

V nástroji **Tools** → **Vegetation Editor** byly vytvořeny 3 skupiny vegetace pomocí tlačítka **Add Group**. Vytvořené skupiny nesou názvy *Grass*, *Bushes* a *Trees*.

Vytvořená skupina *Grass* obsahuje následující objekty *weed_v_var* a *small_grass_a*, které budou tvořit základní pokrytí terénu travnatým porostem s defaultním nastavením parametrů vyjma parametrů, které byly nastaveny následujícím způsobem:

- **Random Rotation** – *True*,
- **Use Terrain Color** – *True*,
- **Air Resistance** – *0,7*,
- **Density** – *6*,
- **Elevation Min** – *0*.

Tato skupina byla nanесena pomocí nástroje **Paint** s nastaveným rádiusem *60*.

Skupina *Bushes* obsahuje následující objekty *green_fern_bush_b* a *bux_hedge_ball_big*, které budou tvořit pokrytí terénu větším porostem s defaultním nastavením parametrů vyjma parametrů, které byly nastaveny následujícím způsobem:

- **Random Rotation** – *True*,
- **Air Resistance** – *0,7*,
- **Density** – *45*,
- **Elevation Min** – *16*.

Tato skupina byla nanесena pomocí nástroje **Paint** s nastaveným rádiusem *60*.

Skupina Trees obsahuje následující objekty *river_tree_d*, *river_tree_c* a *river_tree_b*, které budou tvořit pokrytí terénu stromy s defaultním nastavením parametrů vyjma parametrů, které byly nastaveny následujícím způsobem:

- **Random Rotation** – *True*,
- **Air Resistance** – *1*,
- **Density** – *26*,
- **Elevation Min** – *0*.

Tato skupina byla nanесena pomocí nástroje **Paint** s nastaveným rádiusem 60.

Po nanесení těchto tří skupin na terén bylo provedeno doladění pokrytí terénu jednotlivými skupinami pomocí nástroje **Paint** s různým nastavením rádiusu a pomocí nástroje **Erase** byly objekty vegetace odstraněny z povrchů, kam nepatří.



Obrázek 30. Přidána vegetace do levelu

5.4.2 Vkládání objektů pro terén pomocí nástroje Brush

Dotvoření přírodního ohraničení mapy levelu bylo vytvořeno pomocí nástroje **Create Object** → **Brush**. Tímto nástrojem bylo vloženo 42 entit pro skalní útvary, jedná se o entity:

- *objects/natural/rocks/cliff_rocks/cliff_rock_b_big*,
- *objects/natural/rocks/cliff_rocks/cliff_rock_a_med*,
- *objects/natural/rocks/beach_rocks/beach_rock_med_b_con*.

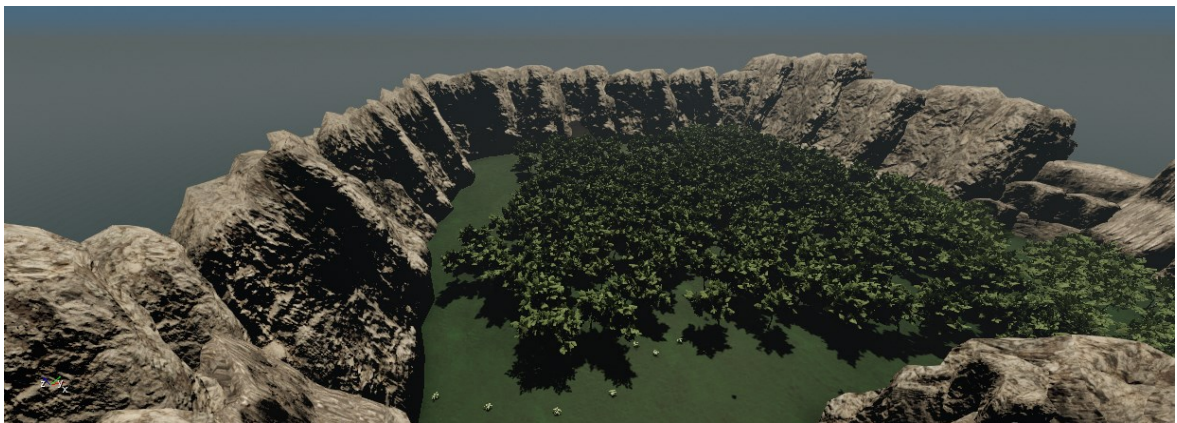
Pro práci s jednotlivými entitami byly dále použity nástroje pro posun – **Move**, rotaci – **Rotate**, velikost – **Scale**, kopírování – **Edit** → **Copy** a vložení – **Edit** → **Paste**.

Pomocí zmíněných nástrojů byly entity rozmístěny po okrajích levelu tak, aby byly vytvořeny hranice pomocí skalních útvarů. Jakmile bylo vytvořeno ohraničení, tak byl level pro-

zkoumán z pohledu první osoby hráče tak, aby byly odhaleny případné nedokonalosti při vkládání a polohování objektů.

Pomocí **Terrain Editor** → **Sculpt** → **Smooth** byly dodělané finální úpravy terénu pro skály tak, aby terén nepřesahoval nad entitami a případně tyto entity nepřekrýval.

Veškeré skalní útvary byly vloženy do nově vytvořené vrstvy **Rocks**, která se nachází v hlavním okně **Level Explorer**. U každé vrstvy je potřeba kliknutím pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru. Všem objektům v položce **Properties** byla zakázána volba **Cast Shadow Maps** pro lepší viditelnost textur a menší nároky na jejich zpracování.



Obrázek 31. Pohled na skalní ohraničení levelu

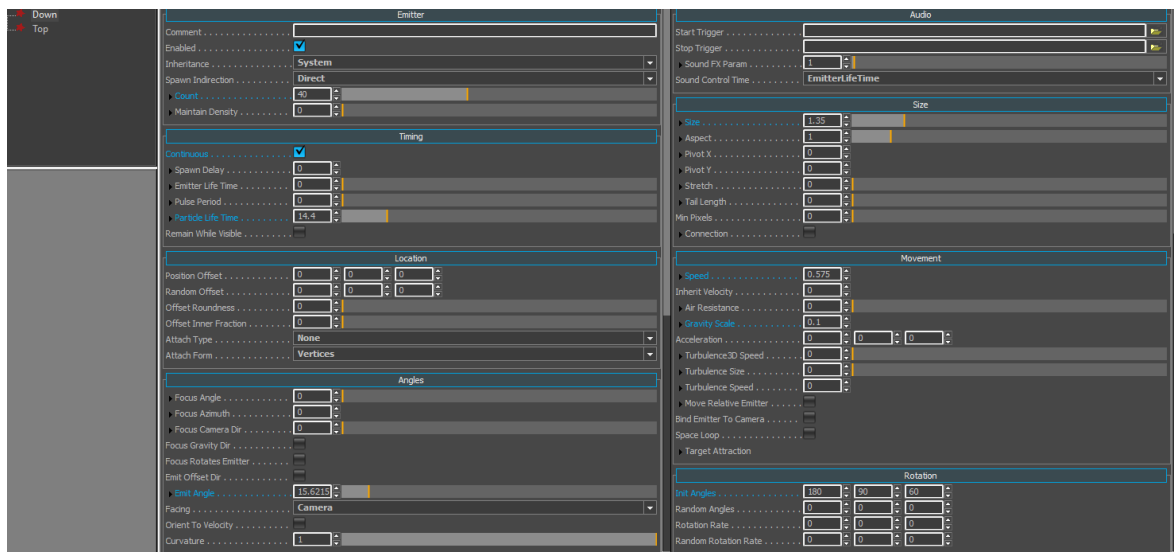
5.4.3 Tvorba vodopádu

Pro samotou texturu vodopádu byl do levelu projektu vložen nástrojem **Create Object** → **Brush** objekt s názvem *objects/natural/rocks/cliff_rocks/cliff_rock_a_med*. Tento objekt byl poté pomocí nástroje **Edit** → **Copy** zkopírován a nástrojem **Move** posunut před původní objekt. Zkopírovanému objektu byla pro efekt padající vody přiřazena nová textura s názvem *objects/natural/water/waterfall/waterfall*.

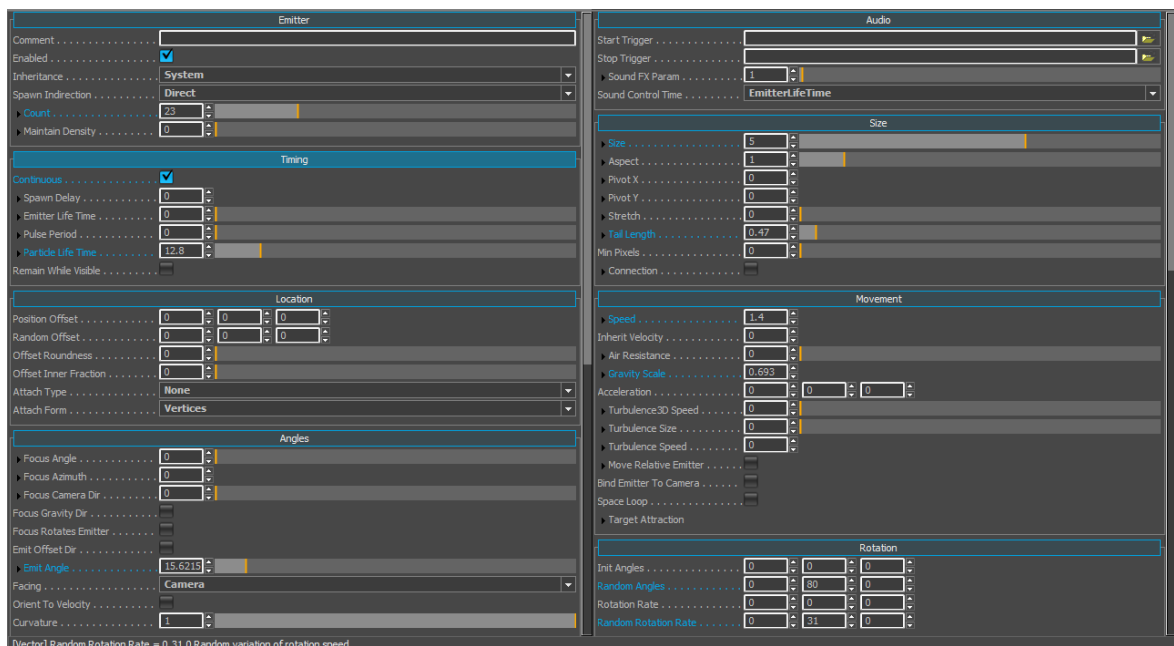
Efekty padající vody byly vytvořeny pomocí objektu **Particle entity** – emitore částic, v nástroji **Tool** → **Database View**. Po kliknutí na záložku **Particles** je zapotřebí vytvořit novou částici pomocí tlačítka **Add New Item**.

Do databáze byly přidány nové částice s názvem **Top** a **Down**. Objekt **Top** představuje emitore částic pro vrchní část vodopádu a **Down** představuje spodní část vodopádu.

Každý emitor má vlastní nastavení pro dosažení požadovaného výsledku tak, jak je zobrazeno v nastavení v obrázcích 32 a 33.



Obrázek 32. Nastavení emitoru pro objekt Down



Obrázek 33. Nastavení emitoru pro objekt Top

Nově vytvořené objekty lze přetáhnutím myši vložit do levelu na požadované místo a pomocí nástrojů **Move**, **Rotate** a **Scale** nastavit polohu, natočení a velikost emitovaných částic do požadované podoby vodopádu. Veškeré objekty pro vodopád byly vloženy do nově vytvořené vrstvy Water, která se nachází v hlavním okně **Level Explorer**. U vrstvy je po-

třeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru.



Obrázek 34. Výsledná podoba vodopádu

5.4.4 Tvorba řeky

Řeka byla vytvořena pomocí nástroje **Create Object** → **Misc** → **River**. Nejprve byla vytvořena křivka, která definuje tvar řeky. Každé kliknutí myši při tvorbě řeky vytváří spoj – čím více spojů, tím složitější tok řeky může být vytvořen. Tuto křivku lze pomocí nástroje **Move** posunout v ose Z níž pod terén, to je vhodné zejména pro tvorbu koryta řeky.

Pomocí funkce **Align Height Map**, která se nachází v **Properties** → **Road**, se terén zarovná v ose Z ke křivce řeky a tím se vytvoří koryto. Příkaz **Properties** → **Road** → **Erase Vegetation** zase odstraní vegetaci v korytě řeky. Řeka byla vytvořena pomocí defaultního nastavení s tím, že parametr řeky **Width** byl nastaven na hodnotu 44 a materiál pro řeku byl nastaven *materials/water/watervolumes/water_forest_b*. Entita řeky poté byla přesunuta do vrstvy Water společně s dalšími dvěma emitory částic (**Down**) pro efekt dopadu vody z vodopádu do řeky.

Pro texturu koryta řeky byla vytvořena další vrstva v **Tools** → **Terrain Editor** → **Paint** s názvem **River** s následujícími parametry:

- **Texture** – Vybrána textura *GameSDK/textures/terrain/river_pepples_no_water*,
- **Material** – Vybrán materiál *GameSDK/materials/terrain/river_bank_terrain_a*.

Tato vrstva byla poté nanášena pomocí nástroje **Paint** na koryto řeky.



Obrázek 35. Výsledná podoba řeky

5.4.5 Tvorba cesty

Nejdříve bylo třeba vytvořit most přes řeku vložení entity mostu, k tomu bylo použito **Create Object** → **Brush** a model *bridges/wooden_bridge/sb_wooden_bridge_main*.

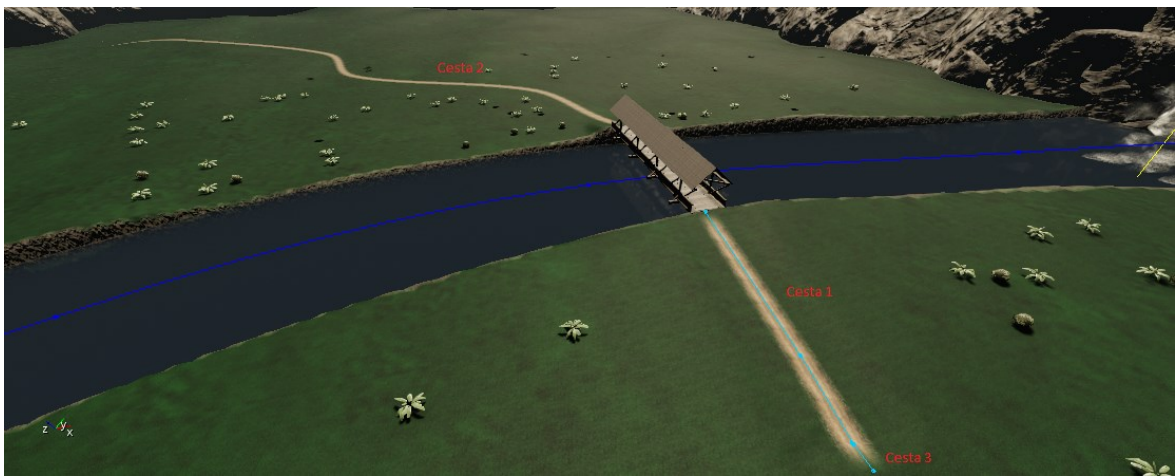
K vytvoření mostu by bylo možné použít jen jednu entitu a pomocí **Scale** upravit délku mostu, tato možnost ovšem nepřipadala v úvahu, protože by most působil nepřirozeným dojmem. Z tohoto důvodu byly použity čtyři entity daného modelu, které byly srovnány do stejné výšky a poté spojeny. Dále bylo úpravou terénu **Tools** → **Terrain Editor** vytvořeno napojení terénu na most. Pro všechny entity mostu byla vypnuta v **Properties** vlastnost **Cast Shadow Maps**.

Veškeré objekty pro most byly vloženy do nově vytvořené vrstvy **Road**, která se nachází v hlavním okně **Level Explorer**. U vrstvy je potřeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru.

Cesta se tvoří podobně jako objekt řeka, ale s nástrojem **Create Object** → **Misc** → **Road**. Cestu tvoří dohromady tři entity cesty. První entita je cesta, která vede z hráčovy startovní lokace k mostu. Druhá cesta vede z mostu k lokaci, kde se bude nacházet cíl úkolu. Třetí cesta je vytvořena kvůli lepšímu přechodu z travnatého terénu k cestě první. Pro každou cestu byla nejprve vytvořena křivka, která definuje její tvar; každé kliknutí myši při tvorbě cesty vytváří spoj – čím více spojů, tím složitější tvar cesty může být vytvořen.

Pomocí funkce **Align Height Map**, která se nachází v **Properties** → **Road**, se terén zarovná v ose **Z** ke křivce cesty a tím dojde k plynulému přechodu z terénu na entitu cesty.

Properties → **Road** → **Erase Vegetation** zase odstraní vegetaci v místě cesty. Cesty byly vytvořeny pomocí defaultního nastavení s tím, že parametr cesty **Width** byl nastaven na hodnotu 4 a materiál pro cesty byl nastaven *materials/roads/forest_trail*. Entity všech tří cest byly poté přesunuty do vrstvy Road. Poté byly prostřednictvím nástrojů editor terénu (**Tools** → **Terrain Editor**) a editor vegetace (**Tools** → **Vegetation Editor**) provedeny finální úpravy v okolí cesty, které byly následně implementovány automatickými funkcemi **Align Height Map** a **Erase Vegetation**. Výsledná cesta je zobrazena na obrázku 36, pro lepší viditelnost cesty byly skryty modely vegetace stromů.



Obrázek 36. Výsledná podoba cesty

5.4.6 Vložení objektů prostředí

K dotvoření celkového prostředí levelu byly použity nástroje **Create Object** → **Brush** a **Create Object** → **Legacy Entities**. Nástrojem **Brush** byla vytvořena malá vesnice, v této vesnici se bude nacházet první cíl mise. Vesnice byla vytvořena z modelů:

- *objects/architecture/fences/wooden_fence/wooden_fence,*
- *objects/architecture/buildings/fishing_houses/wood_table,*
- *objects/architecture/buildings/fishing_houses/wood_chair,*
- *objects/architecture/buildings/outdoor_toilet/outdoor_toilet,*
- *objects/architecture/buildings/observatory_tower_a/observatory_tower_a,*
- *objects/architecture/buildings/loghouse/loghouse_pile,*
- *objects/architecture/buildings/forest_cabin/forest_cabin,*
- *objects/architecture/buildings/fishing_houses/fishing_house_shelter,*
- *objects/architecture/buildings/fishing_houses/fishing_house_a,*
- *objects/architecture/buildings/fishing_houses/barrel,*
- *objects/props/misc/camp_fires/camp_fire_set/camp_fire_a.*

K objektu ohniště – *objects/props/misc/camp_fires/camp_fire_set/camp_fire_a* byl přidán emitor částic pro efekt ohně – **Particle Effect**, s názvem Campfire, který byl vytvořen v **Tool** → **Database View** → **Add New Item**.

Objekty na mapě byly napolohovány nástroji **Move**, **Rotate** a **Scale** a terén pod objekty byl zarovnan nástrojem **Tools** → **Terrain Editor** → **Flatten**.



Obrázek 37. Výsledná podoba vesnice

Dále byl vytvořen hangár s budovami, ve kterém se bude nacházet poslední cíl mise. Tato oblast byla rovněž vytvořena nástrojem **Brush**. Zde byly použity tyto modely:

- *objects/architecture/buildings/hangar/hangar*,
- *objects/architecture/buildings/storage_hall/storage_hall_destroyed*,
- *objects/architecture/buildings/storage_hall/storage_hall*.

I zde bylo použito nástrojů **Move**, **Rotate**, **Scale** a **Tools** → **Terrain Editor** → **Flatten** pro konečné úpravy.

Do modelu hangáru byla vložena okna, která lze rozbít pro lepší efekt – model *objects/props/misc/breakable_glass/window2*. Tyto modely byly vloženy přímo z **Asset Browseru** přetažením myši a byla upravena jejich velikost a umístění tak, aby pasovala do modelu hangáru.

Pro celkový podkres prostředí byly do levelu doplněny další modely pro kameny a na startovní lokaci hráče vložen model poškozeného vrtulníku společně s emitory částic pro efekty kouře – **Particle Effect**, s názvem Smoke, a oheň – **Particle Effect**, s názvem Fire, které byly vytvořeny v **Tools** → **Database View** → **Add New Item**.

Veškeré modely byly vloženy do nově vytvořené vrstvy Misc, která se nachází v hlavním okně **Level Explorer**. U vrstvy je potřeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru.



Obrázek 38. Výsledná podoba objektu cíle

Dále byly do levelu vloženy entity pomocí nástroje **Create Object** → **Legacy Entities**. Tyto entity obsahují různé efekty a spouštěče. Do levelu byly vloženy následující entity pro dokreslení prostředí levelu:

- **Turtles** – želvy jsou umístěny poblíž vodopádu. Entita byla vložena s defaultním nastavením s tím, že byla upravena hodnota parametru **Count** na hodnotu 3 a parametry **Pickable When Alive** a **Pickable When Dead** vypnuty.
- **Frogs** – žáby jsou umístěny na druhé straně koryta řeky u vodopádu. Entita byla vložena s defaultním nastavením s tím, že byla upravena hodnota parametru **Count** na parametry 4 a hodnoty **Pickable When Alive** a **Pickable When Dead** vypnuty.
- **Fish** – ryby jsou umístěny v korytě řeky. Dohromady byly vloženy čtyři entity typu Fish. Entita byly vloženy s defaultním nastavením s tím, že byla upravena hodnota parametru **Count** na hodnotu 20.
- **Chickens** – kohouti jsou umístěni ve vesnici prvního cíle. Entita byla vložena s defaultním nastavením s tím, že byla upravena hodnota parametru **Count** na hodnotu 3 a parametry **Pickable When Alive** a **Pickable When Dead** vypnuty.

- **Bugs** – hmyz je umístěn v lese a podél cesty. Dohromady bylo vloženo dvacet jedna entit typu Bugs. Entity byly vloženy s defaultním nastavením s tím, že byla upravena hodnota parametru **Behaviour** na hodnotu *1* a parametr **Activate** zapnut.
- **Birds** – ptáci jsou umístěni podél první cesty před mostem. Dohromady byly vloženy dvě entity typu Birds. Entity byly vloženy s defaultním nastavením.
- **Light** – osvětlení je umístěno ve vesnici prvního cíle pro přípravu na užití nástroje Lens Flare. Tato entita byla vložena s defaultním nastavením s tím, že byly upraveny hodnoty parametrů **Attenuation** **Buld Size** na hodnotu *5* a **Radius** nastaven na *5*.
- **HealthPack** – lékárničky jsou umístěny ve vesnici prvního cíle a ve startovní lokaci pro doplnění životů hráčovy postavy. Tyto entity byly vloženy s defaultním nastavením s tím, že byly upraveny hodnoty parametrů **Health** na hodnotu *10000* a **Respawn Time** nastaven na *60*.
- **AmmoCrate** – kufr s municí je umístěn ve vesnici prvního cíle a ve startovní lokaci pro doplnění munice do zbraní hráčovy postavy. Tato entita byla vložena s defaultním nastavením s tím, že byly upraveny hodnoty parametrů **Number Of Recharges** na hodnotu *0* a **Use Message** nastaven na „Refill all ammo“.
- **PlayerSpawnPoint** – hráčova počáteční pozice je ve startovní lokaci. Tato entita byla vložena s defaultním nastavením.
- **MH60_Blackhawk** – pozice pro vrtulník Blackhawk se nachází v poslední cílové lokaci. Tato entita byla vložena s defaultním nastavením.

Veškeré modely byly vloženy do nově vytvořené vrstvy Game, která se nachází v hlavním okně **Level Explorer**. U vrstvy je potřeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru.



Obrázek 39. Výsledná startovní lokace

5.4.7 Nastavení prostředí levelu a počasí

Pro přidání efektu mraků je od verze CryEngine 5.4 nutné pro uchování tohoto nastavení přidat do souboru **gamesdk.cryproject** (tento soubor lze otevřít v textovém editoru, například notepad) pod **console_variables** příkaz **"r_VolumetricClouds": "1 []"**, viz obrázek 40.

```

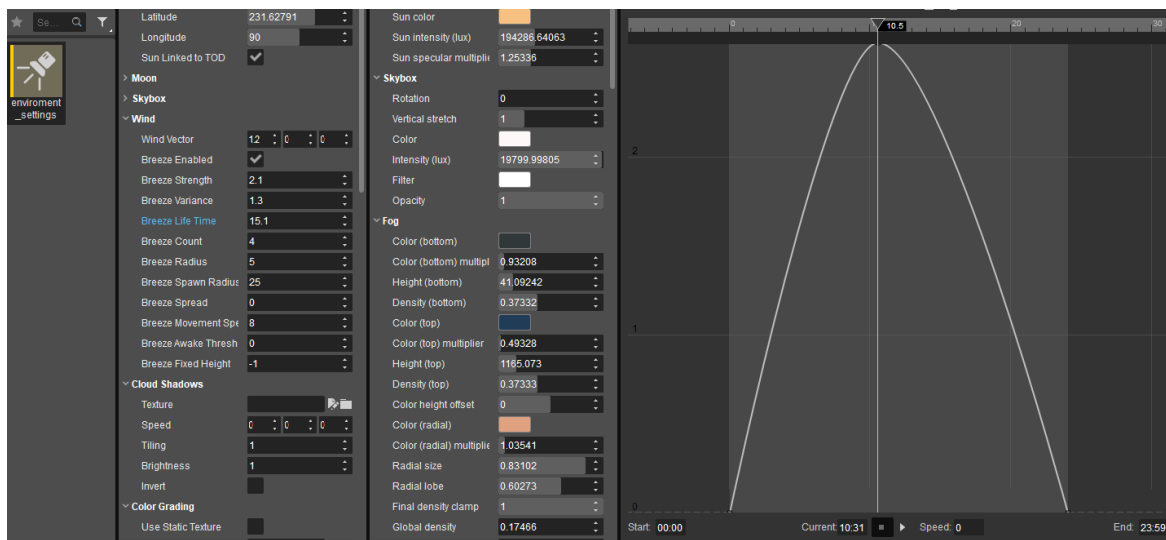
    {
      "type": "EPluginType::Native",
      "path": "CryLobby"
    }
  ],
  "console_variables": {
    "r_VolumetricClouds": "1 [ ]",
    "sys_ime": "1",
    "sys_splashscreen": "Libs/UI/textures/startscreen.tif",
    "sys_target_platforms": "pc,ps4,xboxone,linux",
    "ca_SkeletonEffectsPlayAudioInEngine": "0"
  },
  "console_commands": {}
}

```

Obrázek 40. Vložení příkazu

Pro vytvořený level je zapotřebí vytvořit nastavení pro prostředí levelu, aby bylo možné upravit a nastavit hodnoty pro proměnné prostředí tak, aby šlo toto nastavení uložit a bylo načteno vždy při otevření projektu nebo .exe souboru. Nové nastavení pro prostředí lze vytvořit v okně **Level Settings** v záložce **Environment Presents** kliknutím na ikonu složky. V nově vytvořeném okně **Select Environment** se po kliknutí pravým tlačítkem myši otevře kontextové menu a výběrem **New Asset** → **Environment** se vytvoří nový soubor pro prostředí, které bylo pojmenováno *enviroment_settings*. Po kliknutí na ikonu editace

záložky **Environment Presets** se otevře okno pro editaci dané entity, které nabízí velké množství parametrů pro změnu prostředí levelu. Tato část byla provedena editací parametrů tak, aby výsledný efekt napodobil deštivé počasí, které bude odpovídat následnému zavedení efektů pro déšť. Hlavně byly změněny parametry pro denní dobu, pro vítr, mraky a osvětlení levelu slunečními paprsky, viz obrázek 41. Toto nastavení je nutné uložit pomocí volby **Save**. Výsledný stav oblohy a slunečního svitu je vidět na obrázku 42.



Obrázek 41. Nastavení prostředí



Obrázek 42. Výsledný pohled na oblohu

Efekt padajícího deště byl vytvořen nástrojem **Tools** → **Particle Editor**. Zde byla vybrána particle entita rain s nastavením parametru **Amount** na 0,6 ve funkci **Spawn:Density** v efektu grafu pro tuto entitu. Obě entity byly vloženy do vrstvy Main v okně **Level explorer**.




Obrázek 43. Výsledný efekt deště

5.5 Přidání audio efektů nástrojem ACE

Pro audio bylo v tomto projektu místo defaultního ACE middleware – SDL Mixer použito staženého middleware – Wwise, protože nabízí větší množství zvukových efektů, které budou v projektu použity a které defaultní middleware postrádá. Wwise byl stažen z databáze prostředků. [19] Po stažení je potřeba zkopírovat složku **CryEngine Wwise Project** do složky projektu *gamesdk\audio*.

Pro nastavení Wwise jako výchozího ACE middleware je potřeba přidat do souboru **gamesdk.cryproject** (tento soubor lze otevřít v textovém editoru, například notepad) pod **console_variables** příkaz **"s_ImplName": "CryAudioImplWwise"**, viz obrázek 44.

Dále je důležité v Sandbox Editoru otevřít nástroj **Tools** → **Audio Controls Editor**. Po otevření okna **Audio Controls Editor** je nutné změnit nastavení pro **Project Path** na *audio/CryEngine Wwise Project*, toho lze dosáhnout kliknutím na ikonu  → **Edit** → **Preferences**.


```

        "path": "CryScaleform",
        "type": "EPluginType::Native"
    },
    {
        "type": "EPluginType::Native",
        "path": "CryLobby"
    }
]
},
"console_variables": {
    "s_ImplName": "CryAudioImplWwise",
    "r_VolumetricClouds": "1 [ ]",
    "sys_ime": "1",
    "sys_splashscreen": "Libs/UI/textures/startscreen.tif",
    "sys_target_platforms": "pc,ps4,xboxone,linux",
    "ca_SkeletonEffectsPlayAudioInEngine": "0"
},
"console_commands": {}
}

```

Obrázek 44. Editace gamesdk.cryproject

Po nastavení výchozího ACE middleware je možno vložit více zvukových efektů pro vytvořený level.

Do levelu byly vloženy celkem čtyři entity pro audio efekty nástrojem **Create Object** → **Audio** → **Audio Trigger Spot** s následujícím umístěním a nastavením:

- **AudioWaterfall** – entita umístěna v objektu vodopádu s defaultním nastavením s tím, že parametr **Play Trigger Name** byl nastaven na *Play_l_global_sfx_waterfall_large_loop*.
- **AudioRain** – entita umístěna ve startovní lokaci. Tato entita poskytuje efekt pro celou mapu levelu. Entita byla vytvořena s defaultním nastavením s tím, že parametr **Play Trigger Name** byl nastaven na *Play_l_global_amb_rain_exterior*.
- **AudioFire** – entita umístěna ve startovní lokaci v modelu poškozeného vrtulníku BlackHawk s defaultním nastavením. Parametr **Play Trigger Name** byl nastaven na *Play_p_pfx_fire_blackhawk_mh60*.
- **AudioCampfire** – entita umístěna v první cílové lokaci vesnice v modelu ohniště s defaultním nastavením. Parametr **Play Trigger Name** byl nastaven na *Play_p_pfx_fire_campfire*.

Veškeré entity pro audio efekty byly vloženy do nově vytvořené vrstvy AudioFX, která se nachází v hlavním okně **Level Explorer**. U vrstvy je potřeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru.

5.6 Tvorba cílů hry pomocí Flow graph

Tato kapitola byla tvořena s pomocí knih Mastering CryEngine [2], CryEngine1 Game Programming with C++, C#, and Lua [20] a CryEngine Game Development Blueprints [21]. Tyto knihy jsou věnovány CryEngine 3, tudíž některé informace nebyly aktuální, a tak byly dodatečné informace čerpány z online tutoriálů [18].

Pro vytvoření cílů hry je nejprve zapotřebí vytvořit soubor objectives.xml v umístění `\gamesdk\levels\singleplayer\level\leveldata\objectives.xml` a do něj zapsat úkoly misí.

```
<Root>
  <Level Name="level">
    <Mission_01 Name="Find fuel." Description="Find fuel for
helicopter."/>
    <Mission_02 Name="Take fuel." Description="Take fuel for
helicopter."/>
    <Mission_03 Name="Use fuel." Description="Fill fuel to the
helicopter."/>
    <Mission_04 Name="Take helicopter" Description="Leave this
place with helicopter."/>
  </Level>
</Root>
```

Obrázek 45. Vytvořené úkoly v xml souboru

V nástroji **Tools** → **Database View** v záložce **Game Tokens** byly vytvořeny proměnné pro jednotlivé stavy misí.

| Name | Value | Type |
|-----------------------|-------|------|
| Group: | | |
| Find_fuel_activated | false | Bool |
| Take_fuel_activated | false | Bool |
| Use_Heli_activated | false | Bool |
| Use_Heli_deactivated | false | Bool |
| Use_Heli_completed | false | Bool |
| Take_fuel_deactivated | false | Bool |
| Find_fuel_deactivated | false | Bool |
| Find_fuel_completed | false | Bool |
| Take_fuel_completed | false | Bool |
| Fill_heli_activated | false | Bool |
| Fill_heli_deactivated | false | Bool |
| Fill_heli_completed | false | Bool |

Obrázek 46. Vytvořené proměnné pro stavy úkolů

Dále bylo vytvořeno nástrojem **Create Object** → **Legacy Entities** osm entit

Následujícího názvu, typu a účelu:

- **AreaTrigger-1** – AreaTrigger – spouštěcí entita pro první misi,
- **AreaTrigger-2** – AreaTrigger – spouštěcí entita pro třetí misi,
- **FillFuel** – Switch – tlačítko pro splnění cíle druhé mise (zde byl změněn výchozí model tlačítka switch),
- **Fuel** – Switch – tlačítko pro splnění cíle třetí mise,
- **Mission1** – MissionObjective – entita reprezentující první misi,
- **Mission2** – MissionObjective – entita reprezentující druhou misi,
- **Mission3** – MissionObjective – entita reprezentující třetí misi,
- **Mission4** – MissionObjective – entita reprezentující čtvrtou misi.

Dále byly vytvořeny nástrojem **Create Object** → **Area** → **Box** dvě entity následujícího názvu, typu a účelu:

- **AreaBox-1** – AreaBox – Ohraničení pro spouštěcí entitu první mise,
- **AreaBox-2** – AreaBox – Ohraničení pro spouštěcí entitu třetí mise.

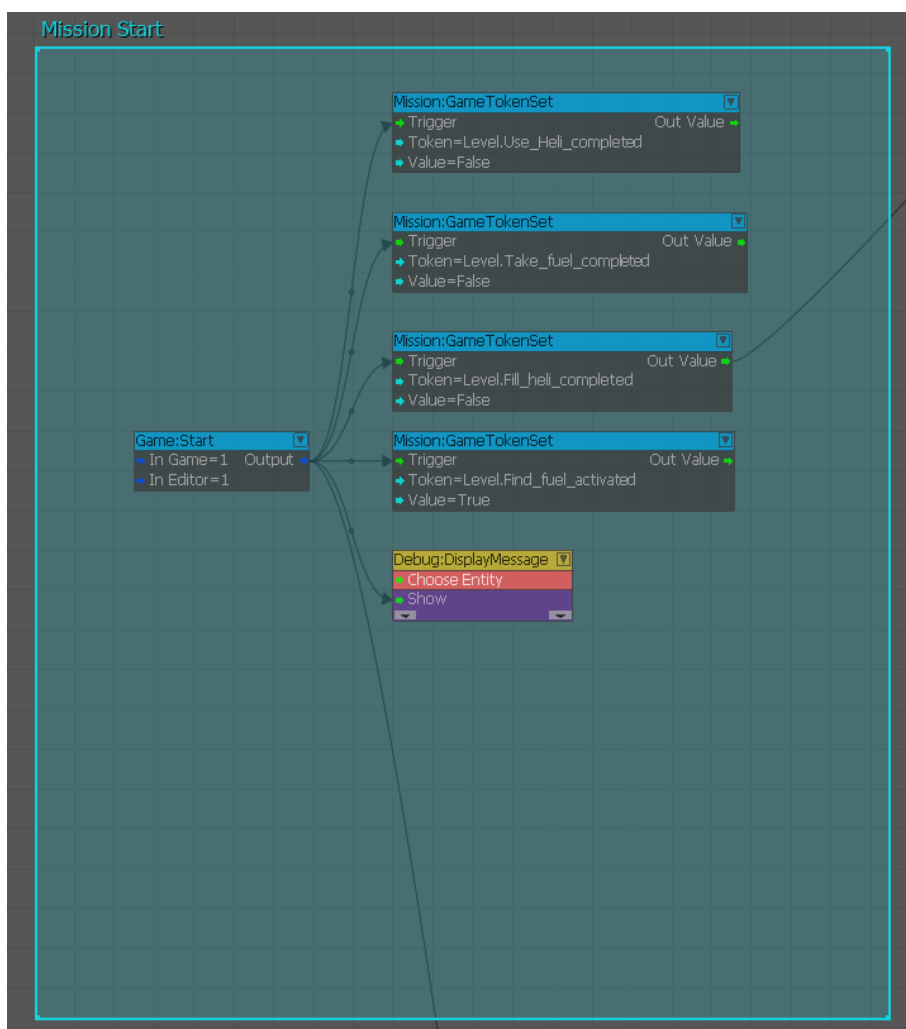
Na závěr byla vytvořena nástrojem **Create Object** → **Empty Entity** entita s názvem **Missions**, která obsahuje Flow graph s logikou implementace pro systém dosažení cílů hry.

Veškeré entity byly vloženy do nově vytvořené vrstvy **Objectives**, která se nachází v hlavním okně **Level Explorer**. U vrstvy je potřeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru.

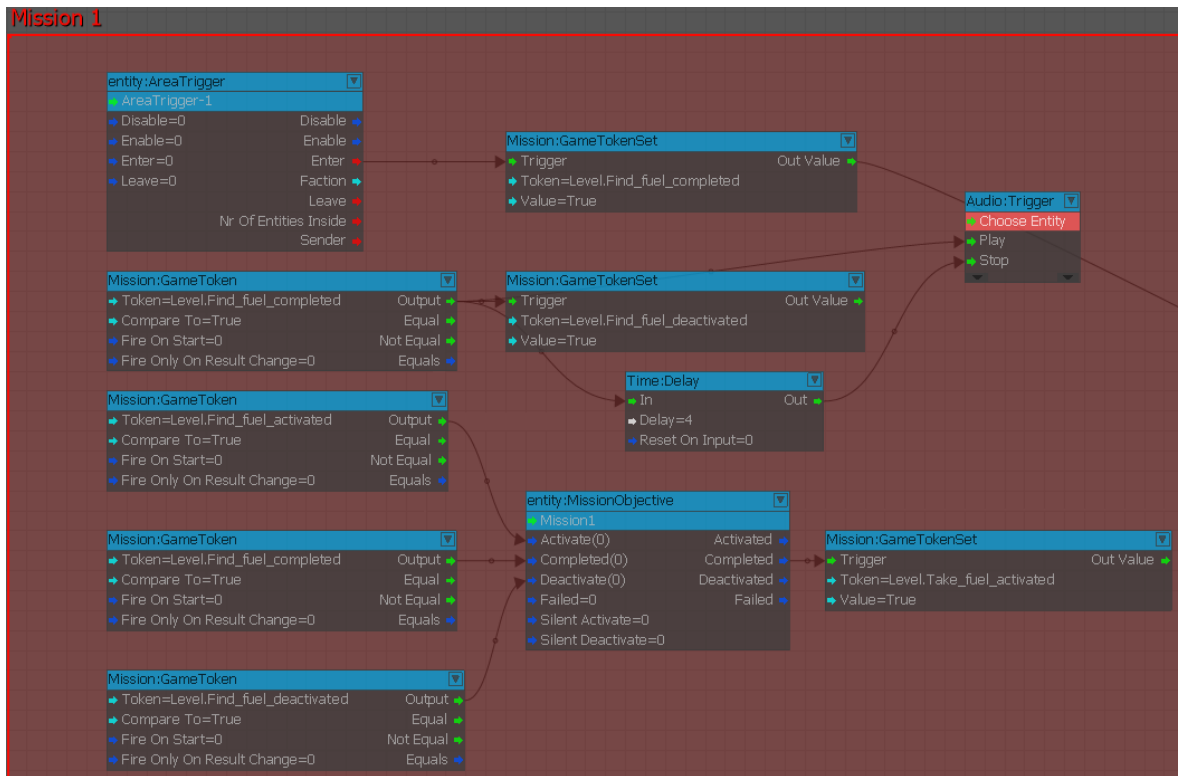
Celý systém cílů hry je realizován následujícím způsobem. Hráči se po načtení levelu zobrazí text, který jej bude informovat o krátkém příběhu. Na HUD se hráči zobrazí první cíl mise, a to najít palivo do vrtulníku. Palivo se nachází v cisterně ve vesnici v pozici prvního cíle hry, který je definován spouštěčem **AreaTrigger-1**. Po nalezení cisterny se první úkol dokončí a zobrazí se druhý úkol. Druhým úkolem je, aby hráč načerpal palivo pro vrtulník. Tento cíl hráč splní dvojným stisknutím klávesy **F** u cisterny – entita **FillFuel**, jakmile se zobrazí zpráva „Take fuel“, poté se cíl dokončí a v HUD se zobrazí hráči třetí cíl hry. Třetím cílem hry je palivo doplnit do vrtulníku. Tento úkol lze splnit u hangáru dvojným stisknutím klávesy **F**, po zobrazení zprávy „Fill fuel“ na tlačítku entity **Fuel**, která se nachází na vrtulníku. Poté se zobrazí čtvrtá mise, která hráče informuje, že nyní může odletět

s vrtulníkem. Po nástupu do vrtulníku je mise splněna, příběh končí a aplikace je ukončena. Hráč si může aktuální misi zobrazit stisknutím klávesy **O**. V cílech hry jsou ošetřeny situace, kdy hráč chce použít vrtulník, aniž by doplnil palivo. Zobrazí se upozornění, které informuje hráče o tom, že vrtulník potřebuje palivo a není schopen s vrtulníkem odletět – úkol čtyři. Stejně tak je ošetřena situace, kdy hráč chce do vrtulníku natankovat palivo, ale palivo nenabral. Zobrazí se upozornění, že nemá palivo a není mu umožněna interakce s tlačítkem úkol tři. Úkol dva lze splnit pouze tehdy, pokud se hráč dostane do vymezeného prostoru entity **AreaTrigger-1**. Úkol jedna se spouští hned při načtení levelu.

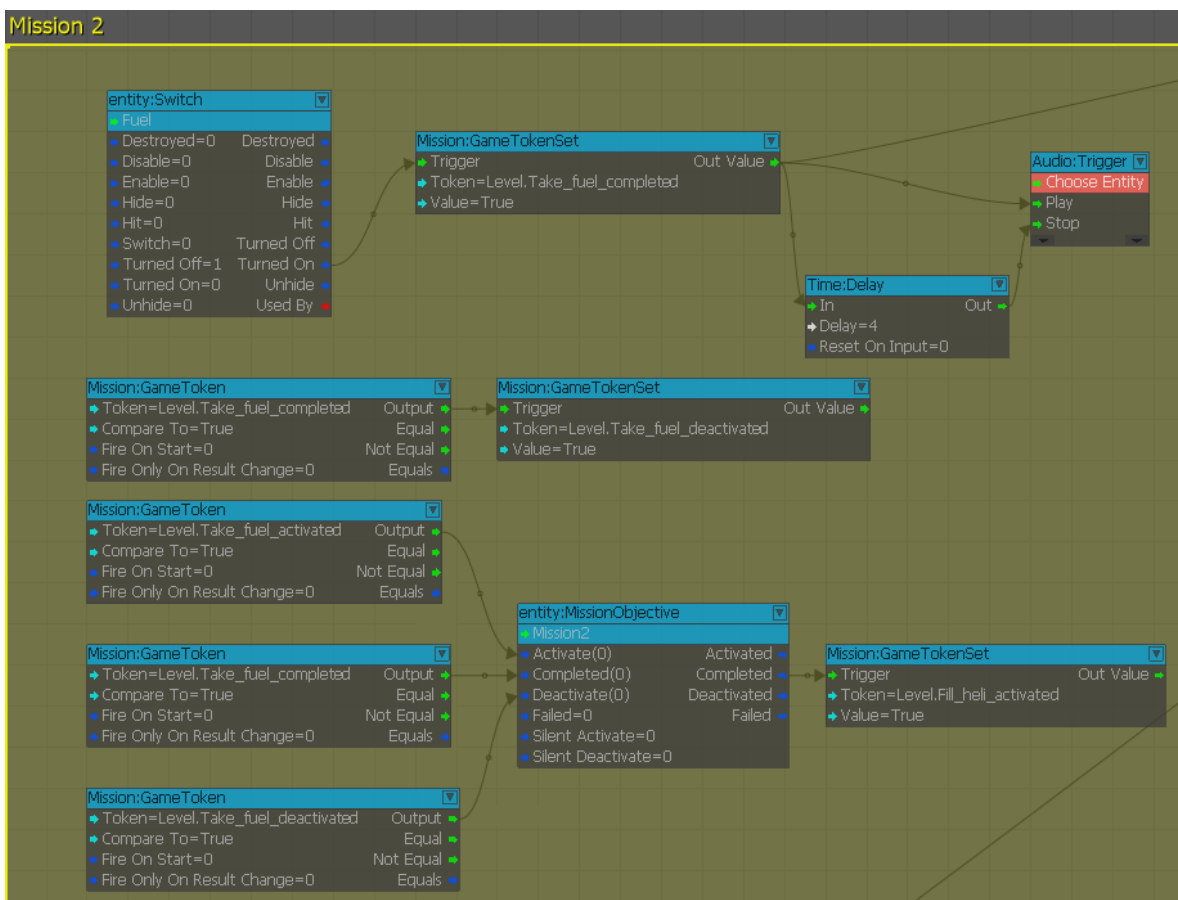
Následující obrázky znázorňují implementaci logiky úkolů pomocí nástroje **Tools** → **Flow Graph**.



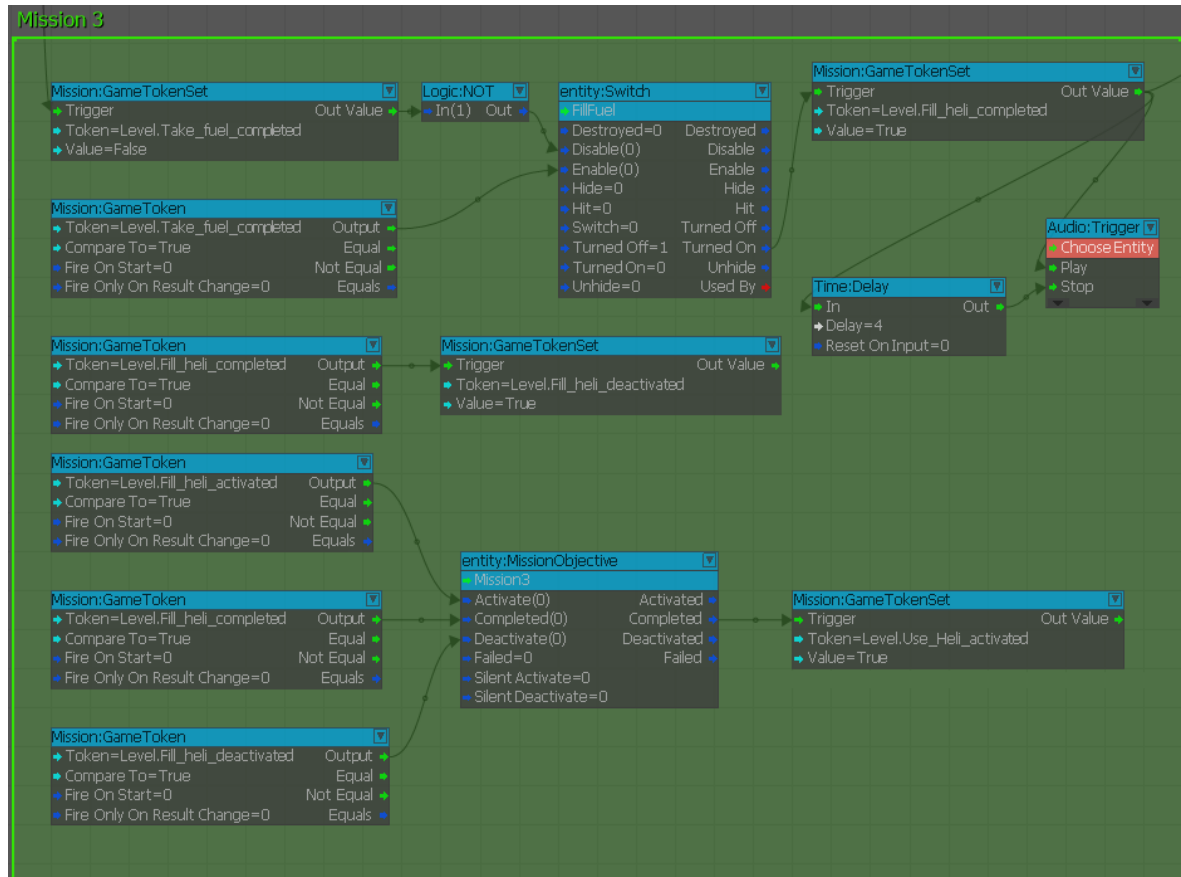
Obrázek 47. Inicializace proměnných při načtení levelu



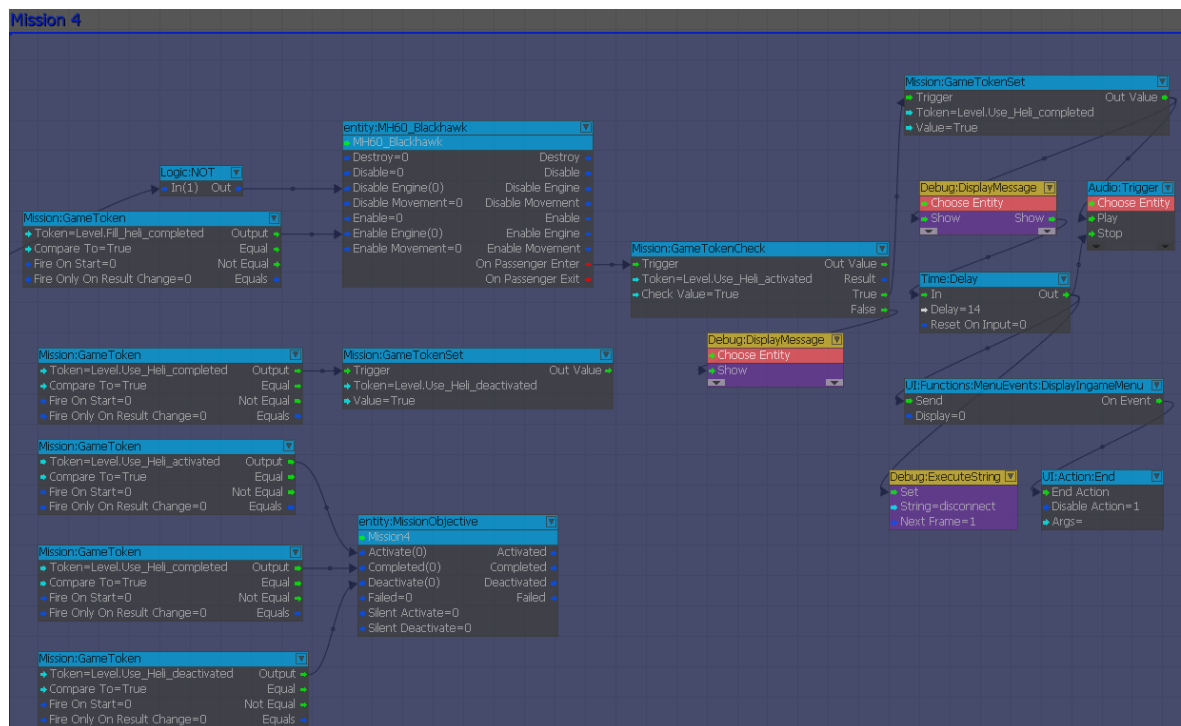
Obrázek 48. Implementace prvního úkolu



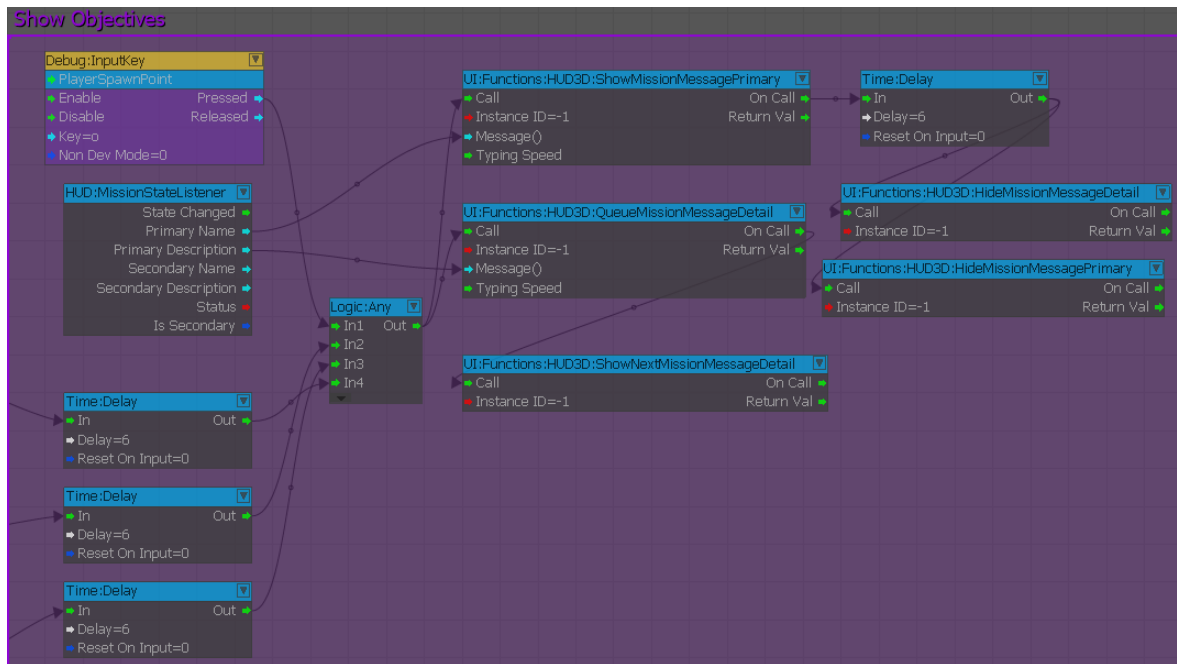
Obrázek 49. Implementace druhého úkolu



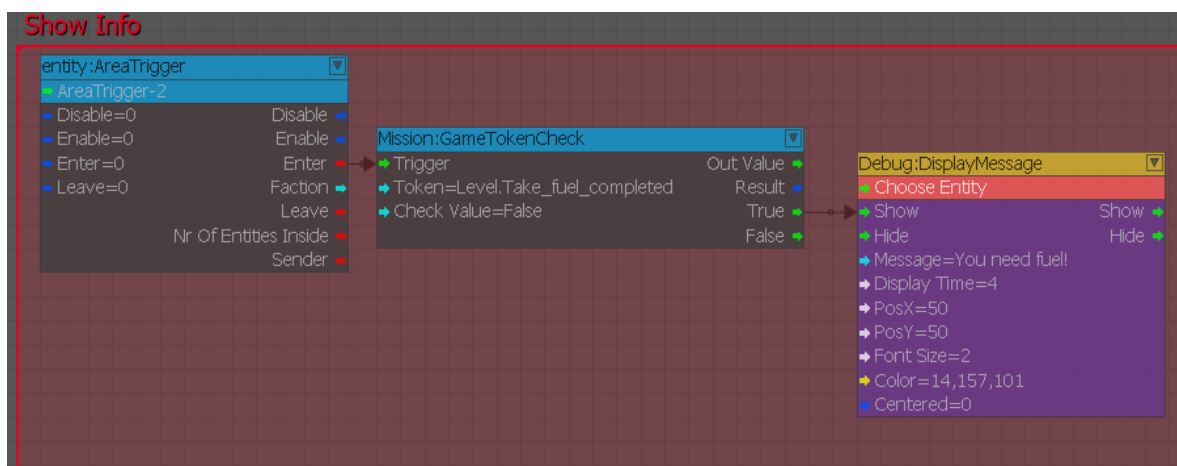
Obrázek 50. Implementace třetího úkolu



Obrázek 51. Implementace čtvrtého úkolu



Obrázek 52. Implementace zobrazení aktuálního úkolu



Obrázek 53. Implementace upozornění

5.7 Úprava hlavního menu a obtížnosti

Pomocí nástroje **Tools** → **Flow Graph** bylo rovněž upraveno hlavní menu, které ukázkový projekt poskytuje. Z menu byla odstraněna tlačítka pro multiplayer, připojení k serveru a jiné. Tlačítko **Single player** bylo nahrazeno tlačítkem **Play**. Byly rovněž odstraněny obtížnosti vyšší než normal. Výběr obtížnosti nyní nabízí obtížnost **Easy** a **Normal**. V adresáři **gamesdk** byla vytvořena složka **Difficulty**, ve které byly vytvořeny dva soubory *easy.cfg* a *normal.cfg*, které odpovídají zvolené obtížnosti. Hlavní rozdíl ve zvolené obtížnosti je ten, že v obtížnosti Easy se hráči po pěti sekundách doplní automaticky životy

na plný počet. U obtížnosti Normal se tomu tak neděje a hráč je tedy odkázán pouze na lékárníčku, kterou může najít na mapě ve vesnici a ve startovní lokaci.

5.8 Použití nástroje Lens Flare na objekt Slunce

Pro vytvoření efektu odlesku objektivu ze slunečního svitu je nejprve důležité vložit do mapy levelu entitu typu **Light** nástrojem **Create Object** → **Legacy Entities**. Na umístění této entity na mapě nezáleží, po aplikaci efektu **Lens Flare** tato entita již nebude vydávat světlo. V **Properties** entity Light, která byla dříve vytvořena ve vesnici prvního cíle, byl zapnut parametr **Flare Enable**. Dále byl použit nástroj **Tools** → **Lens Flare Editor**. Po zobrazení okna Lens Flare Editoru, byla tlačítkem **Load Library** přidána knihovna `sample_flares`. Pro efekt odlesku byl vybrán efekt `generic.generic_sun_flare`. Po označení entity Light v Sandbox Editoru byl v okně Lens Flare Editoru aplikován efekt tlačítkem **Assign Item to Selected Objects**. Pro aplikaci efektu na Slunce byl nastaven v **Properties** entity **Light** parametr **Diffuse Multiplier** na hodnotu 10 a byl zapnut parametr **Attach to Sun**.



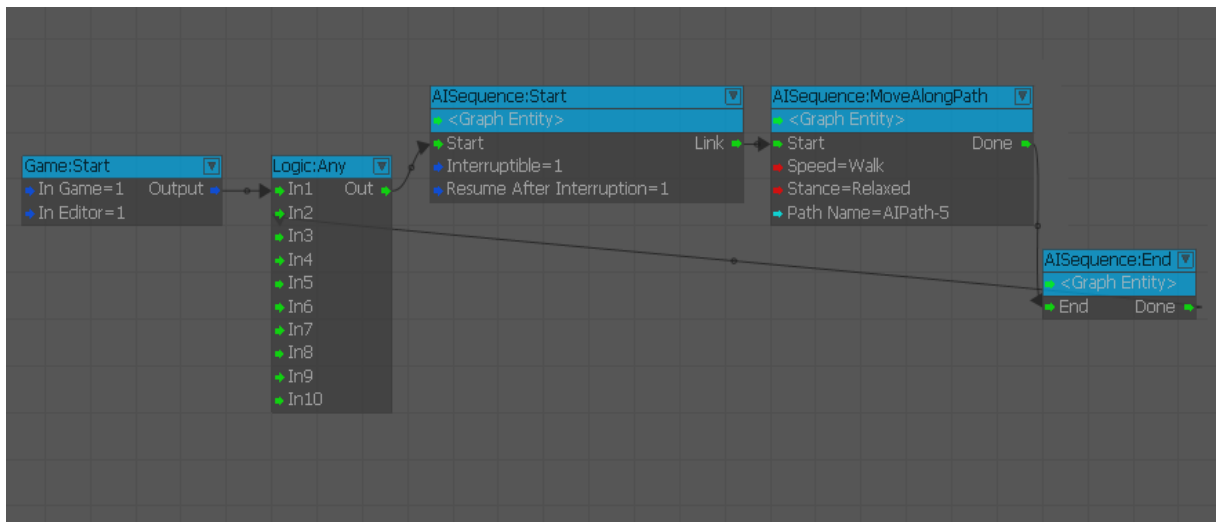
Obrázek 54. Výsledný efekt Lens Flare

5.9 Vložení a nastavení nepřátelských postav

Do levelu bylo vloženo nástrojem **Create Object** → **Legacy Entities** celkem 47 objektů typu **Human**. Tyto objekty byly vloženy s defaultním nastavením s tím, že menším skupinám objektů byl změněn parametr **Model** tak, aby nepřátelé ve hře nevypadali stejně. Po testování herního prostředí a reakce nepřátel byly změněny parametry u všech nepřátel tak, aby úhel zorného pole, vzdálenost detekce hráče a reakce nepřátel odpovídaly prostředí a náročnosti hry s ohledem na to, aby hra byla dostatečně náročná, ale přitom nebyla příliš obtížná. Tímto způsobem byly změněny následující parametry:

- **Auto Disable** – Zapnuto,
- **Go Back To Start On Idle** – Zapnuto,
- **Interest** → **Angle** – 270,
- **Min Interest Level** – 0,
- **FOV Primary** – 180,
- **FOV Secondary** – 270,
- **Sightrange** – 35.

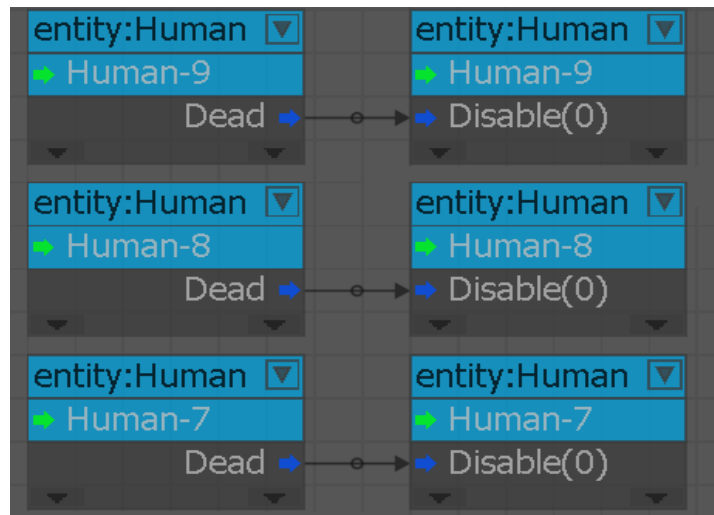
Dále bylo do levelu vloženo nástrojem **Create Object** → **AI** → **AI Path** celkem 16 objektů typu **AIPath**. Tyto objekty definují cestu, po které se budou nepřátelé pohybovat. Každý nepřítel, kterému je přiřazena cesta, má vlastní **Flow Graph** s implementovanou logikou pro sledování této cesty, viz. obrázek 55.



Obrázek 55. Implementace logiky pro sledování cesty

Nakonec byla do levelu nástrojem **Create Object** → **Empty Entity** vložena prázdná, pro hráče neviditelná entita, která obsahuje **Flow Graph** logiku pro odstranění objektu **Human** ze hry po jeho eliminaci, viz. obrázek 56.

Veškeré nové entity byly vloženy do nově vytvořené vrstvy **Enemies**, která se nachází v hlavním okně **Level Explorer**. U vrstvy je potřeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru.



Obrázek 56. Odstranění modelu nepřítele

5.10 Finální úpravy

V této fázi byla hra poskytnuta dobrovolníkům k otestování hratelnosti a celkového dojmu z vytvořeného levelu. Na základě testování byly od dobrovolníků poskytnuty tyto podněty k vylepšení:

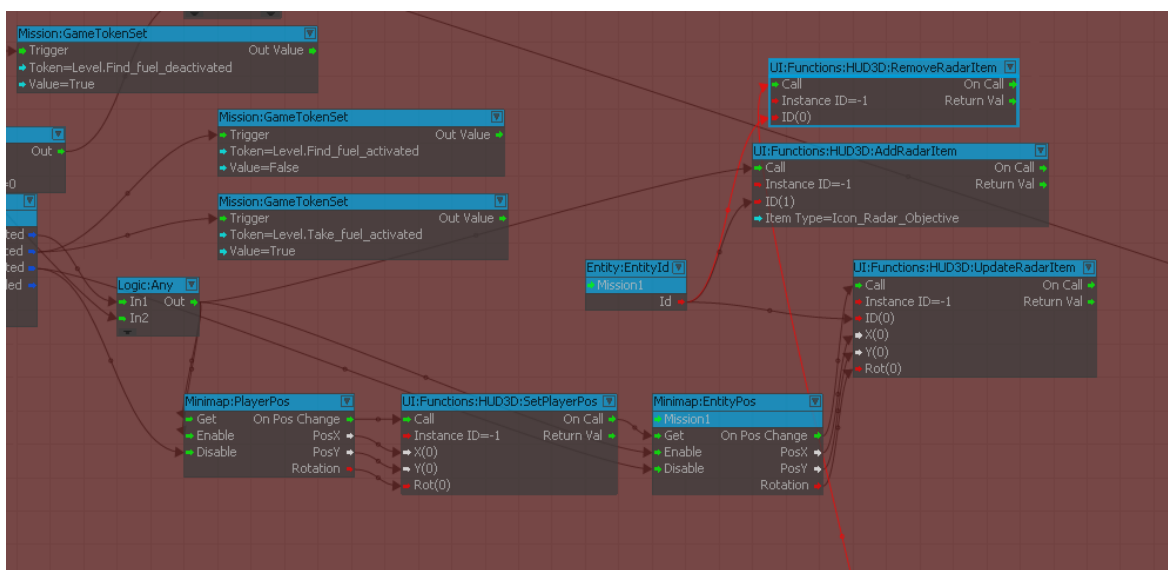
- Vylepšit celkové osvětlení levelu.
- Příkré břehy řeky – upravit břehy řeky tak, aby se hráč po pádu do řeky mohl dostat na břeh.
- Absence zobrazení kláves pro ovládání hry.
- Přidat navigaci k cílům mise.

Do mapy levelu bylo přidáno celkem 38 entit typu **Light** pro celkové osvětlení mapy. Každá entita má umístění a parametry nastavené tak, aby byla mapa lépe osvětlena, zvláště pod stromy. Tyto entity byly vloženy do nově vytvořené vrstvy MapLights, která se nachází v hlavním okně **Level Explorer**. U vrstvy je potřeba po kliknutí pravým tlačítkem myši v kontextovém menu v položce **Layer** → **More** vybrat možnost **Auto-Load** pro zobrazení této vrstvy ve finálním .exe souboru. Zároveň byly upraveny parametry prostředí levelu pro lepší světelný efekt.

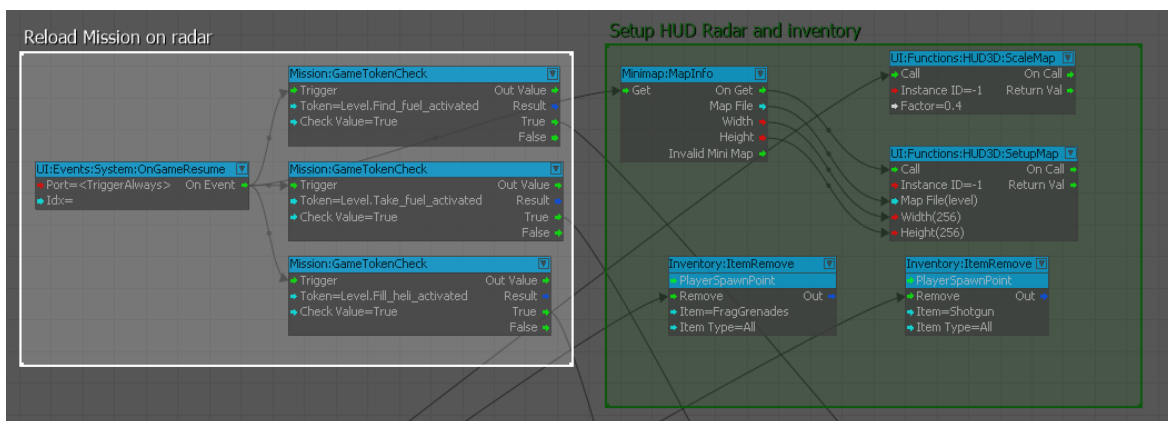
Břehy řeky byly upraveny pomocí Terrain Editoru dle požadavků a zároveň bylo přidáno více trávy a keřů do levelu pro lepší vizuální efekt.

Do hlavního menu a do menu při pozastavení hry bylo tlačítko **Show Controls**, které načte nově vytvořené menu v nástroji **Flow Graph**. Toto menu zobrazuje přiřazení kláves k možným akcím ve hře.

Na herním HUD byla aktivována funkce navigace k aktuálnímu cíli mise, kterou reprezentuje červená tečka v levém dolním rohu obrazovky. Logika navigace byla implementována v již existujícím **Flow Graph** grafu s názvem Missions pro každý úkol mise zvlášť, viz obrázek 57. Tato část byla problémová. Například po vyvolání menu pozastavením hry a návratu do hry došlo ke ztrátě údajů o pozici hráče a pozici úkolu a již nedošlo k opětovnému načtení údajů. Tento problém je vyřešen na obrázku 58.



Obrázek 57. Doplnění funkcionality navigace k cíli první mise



Obrázek 58. Opětovné načtení údajů aktuální mise po pozastavení hry

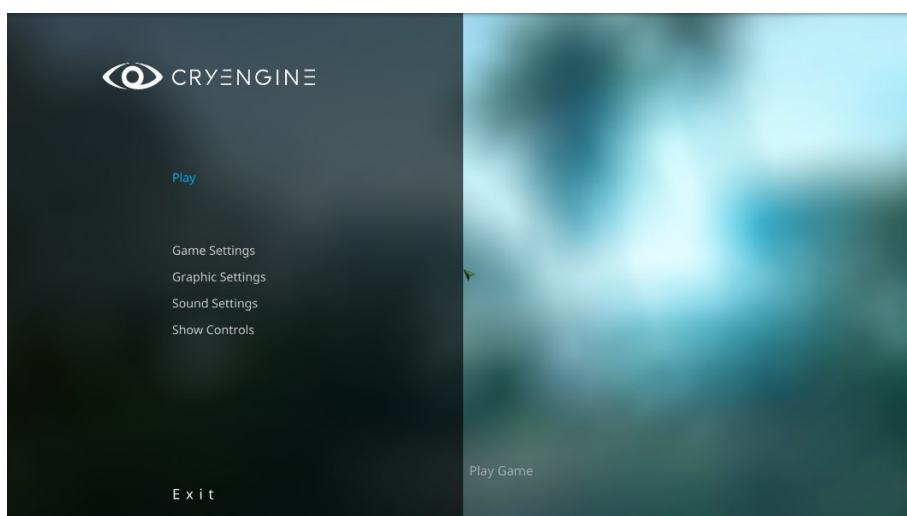
Na závěr pro odstranění in-game debug informací na obrazovce je potřeba přidat do souboru **gamesdk.cryproject** (tento soubor lze otevřít v textovém editoru, například notepad) pod **console_variables** příkaz **"r_DisplayInfo": "0"**, viz obrázek 59.

```
    },  
    {  
      "type": "EPluginType::Native",  
      "path": "CryLobby"  
    }  
  ]  
},  
"console_variables": {  
  "s_ImplName": "CryAudioImplWwise",  
  "r_VolumetricClouds": "1 [ ]",  
  "r_DisplayInfo": "0",  
  "sys_ime": "1",  
  "sys_splashscreen": "Libs/UI/textures/startscreen.tif",  
  "sys_target_platforms": "pc,linux",  
  "ca_SkeletonEffectsPlayAudioInEngine": "0"  
},  
"console_commands": {}  
}
```

Obrázek 59. Deaktivace in-game debug informací

5.11 Uživatelský pohled na hru

Hlavní menu hry zobrazené na obrázku 60 obsahuje tlačítko **Play**, které po kliknutí nabízí volbu obtížnosti a následné spuštění levelu. Tlačítko **Game Settings** nastavuje citlivost myši. Tlačítko **Graphic Settings** umožňuje nastavení celé obrazovky, rozlišení a kvality obrazu. **Sound Settings** nastavuje hlasitost hry a tlačítko **Show Controls** zobrazuje přiřazení kláves k interakcím ve hře. Celostránkové obrázky z této hry jsou umístěny v příloze č. II.



Obrázek 60. Hlavní menu

ZÁVĚR

Cílem této bakalářské práce bylo popsat a představit hlavní vlastnosti a možnosti vývojového prostředí CryEngine (verze 5.7.1 LTS), vytvořit dokumentaci v českém jazyce a vyvinout komplexní aplikaci, ve které bude představena široká škála funkcí a možností CryEngine.

V teoretické části bakalářské práce jsou popsány hlavní vlastnosti a možnosti herního engine CryEngine. Teoretická část dále pojednává o přidaných novinkách ve verzi CryEngine 5.7.1 LTS. Na závěr teoretické části je CryEngine porovnáván s konkurenčním softwarem Unity a Unreal Engine.

Praktická část bakalářské práce je věnována tvorbě dokumentace v českém jazyce a poté se zaměřuje na návrh a tvorbu aplikace od základní myšlenky až po samotnou tvorbu, jejíž proces je v práci podrobně popsán.

Tvorbu aplikace komplikovaly drobné problémy, zejména pak občasné a nečekané pády herního engine při tvorbě mapy levelu. Z tohoto důvodu byl postup při tvorbě aplikace průběžně ukládán a celý projekt byl zálohován. Záloha projektu byla využita při jednom z pádů engine, kdy nešel projekt nadále otevřít a pracovat na něm.

Výstupem této bakalářské práce je tedy manuál v českém jazyce a aplikace. Výsledná aplikace je FPS hra, která demonstruje grafické možnosti tohoto herního engine při použití co největšího množství nástrojů, které CryEngine nabízí.

Součástí této bakalářské práce je USB flash disk. Na disku se nachází CryEngine projekt, ve kterém aplikace vznikala, dokumentace v českém jazyce a samotná aplikace. Projekt lze otevřít v CryEngine otevřením souboru gamesdk.cryproject, který se nachází ve složce Project. Aplikaci lze spustit souborem SWIGame.exe, který se nachází ve složce SWIGame\bin\win_x64. Na disku je rovněž instalátor Visual C++ Redistributable, který je potřebný pro spuštění aplikace.

Projekt i aplikace měly být původně distribuovány pomocí instalátoru na USB flash disku. Instalátor byl tvořen v Microsoft Visual Studiu pomocí rozšíření Microsoft Visual Studio Installer Projects. Po nainstalování aplikace pomocí instalátoru se aplikace sice otevřela, ale již nedošlo k načtení levelu hry. Proto bylo od distribuce pomocí instalátoru upuštěno.

SEZNAM POUŽITÉ LITERATURY

- [1] Crytek - video game developer, makers of CryEngine. *Crytek* [online]. Frankfurt, 2023 [cit. 2023-01-14]. Dostupné z: <https://www.crytek.com/>
- [2] GUNDLACH, Sascha a Michelle K. MARTIN. Mastering CryEngine. In: *Mastering CryEngine*. Birmingham B3 2PB: Packt Publishing, 2014, ISBN 978-1-78355-025-8.
- [3] Games. *Crytek* [online]. Frankfurt, 2023 [cit. 2023-01-14]. Dostupné z: <https://www.crytek.com/games>
- [4] Licensing. *CryEngine* [online]. 2023 [cit. 2023-01-14]. Dostupné z: <https://www.CryEngine.com/support/view/licensing>
- [5] CryEngine: Visuals [online]. 2023 [cit. 2023-01-14]. Dostupné z: <https://www.CryEngine.com/features/view/visuals>
- [6] CryEngine: Sandbox [online]. 2023 [cit. 2023-01-14]. Dostupné z: <https://www.CryEngine.com/features/view/sandbox>
- [7] CryEngine: AI & Animation [online]. 2023 [cit. 2023-01-20]. Dostupné z: <https://www.CryEngine.com/features/view/ai-animation>
- [8] CryEngine: Audio [online]. 2023 [cit. 2023-01-20]. Dostupné z: <https://www.CryEngine.com/features/view/audio>
- [9] CryEngine: Physics [online]. 2023 [cit. 2023-01-20]. Dostupné z: <https://www.CryEngine.com/features/view/physics>
- [10] CryEngine: *Performance* [online]. 2023 [cit. 2023-01-20]. Dostupné z: <https://www.CryEngine.com/features/view/performance>
- [11] CryEngine 5.7. *CryEngine V Manual* [online]. [cit. 2023-01-28]. Dostupné z: <https://docs.CryEngine.com/display/RN/CryEngine+5.7>

- [12] LASTBITCODER. *Geeksforgeeks: Unity vs CryEngine* [online]. 30. srpen 2022 [cit. 2023-01-28]. Dostupné z:
<https://www.geeksforgeeks.org/unity-vs-CryEngine/>
- [13] LASTBITCODER. *Geeksforgeeks: Unreal vs CryEngine1* [online]. 6. červenec 2022 [cit. 2023-01-28]. Dostupné z:
<https://www.geeksforgeeks.org/unreal-vs-CryEngine1/>
- [14] *CryEngine V Manual: CryEngine V Manual* [online]. CRYTEK, 2019 [cit. 2023-02-25]. Dostupné z: <https://docs.CryEngine1.com/>
- [15] *ASSET DATABASE: CryEngine GameSDK Sample Project* [online]. Crytek, 2023 [cit. 2023-04-25]. Dostupné z:
<https://www.CryEngine1.com/marketplace/product/crytek/CryEngine1-gamesdk-sample-project>
- [16] TRACY, Dan a Sean TRACY. *CryEngine1 3 Cookbook*. Birmingham B3 2PB: Packt Publishing, 2011. ISBN 978-1-849691-06-2.
- [17] *TUTORIALS: Game & Level Design* [online]. 2023 Crytek [cit. 2023-03-25]. Dostupné z: TUTORIALS: Game & Level Design [online]. 2023 Crytek [cit. 2023-04-25]. Dostupné z:
<https://www.CryEngine1.com/tutorials/view/game-level-design>
- [18] *GAMEDEV4FUN. CryEngine1 5 Tutorials. CryEngine1 5 Tutorials* [online]. [cit. 2023-04-25]. Dostupné z:
<https://www.youtube.com/playlist?list=PLYnbKITbqv7BgLbTynsBwmOfk8pfiGpei>
- [19] *Asset database: CryEngine Wwise Project* [online]. Crytek, 2023 [cit. 2023-04-25]. Dostupné z:
<https://www.CryEngine1.com/marketplace/product/crytek/CryEngine1-wwise-project>

- [20] LUNDGREN, Filip a Ruan PEARCE-AUTHERS. *CryEngine1 Game Programming with C++, C#, and Lua*. Birmingham B3 2PB, UK: Packt Publishing, 2013. ISBN 978-1-84969-590-9.
- [21] Richard Gerard MARCOUX III, Chris GOODSWEN, Riham TOULAN a Sam HOWELS. *CryEngine Game Development Blueprints*. Birmingham B3 2PB, UK: Packt Publishing, 2015. ISBN 978-1-78439-987-0.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

| | |
|--------|---|
| CGI | computer-generated imagery |
| API | Application Programming Interface |
| PN | Point-Normal |
| SSR | Screen Space Reflections |
| HDR | High Dynamic Range |
| SVOGI | Sparse voxel octree global illumination |
| IBL | Image-based lighting |
| SSDO | Screen Space Directional Occlusion |
| POM | Parallax Occlusion Mapping |
| OBM | Offset Bump Mapping |
| MNM | Multi-Layer Navigation Mesh |
| CCD IK | Cyclic Coordonate Descent Inverse Kynematic |
| ACE | Audio Controls Editor |
| RTCP | Real-Time Transport Control Protocol |
| HRTF | head-related transfer function |
| DRS | Dynamic Response Systém |
| SDK | Software development kit |
| LTS | Long term support |
| HUD | Heads-up display |
| FPS | First person shooter |

SEZNAM OBRÁZKŮ

| | |
|--|----|
| Obrázek 1. Oblastní světla [5] | 12 |
| Obrázek 2. Vykreslování [5]..... | 13 |
| Obrázek 3. Vodní kaustika [5]..... | 14 |
| Obrázek 4. Teselace [5] | 14 |
| Obrázek 5. HDR odlesky objektivu [5] | 15 |
| Obrázek 6. Efektivní antialiasing [5]..... | 16 |
| Obrázek 7. Rozmazání pohybu [5] | 16 |
| Obrázek 8. Místní reflexe [5]..... | 17 |
| Obrázek 9. Vegetace [5] | 18 |
| Obrázek 10. Mapy stínů [5]..... | 19 |
| Obrázek 11. Směrová okluze [5] | 19 |
| Obrázek 12. Osvětlení na základě obrazu [5] | 20 |
| Obrázek 13. Systém částicového efektu [5]..... | 21 |
| Obrázek 14. Objemové stíny mlhy [5]..... | 21 |
| Obrázek 15. Editor materiálů [6] | 22 |
| Obrázek 16. Podpora FBX [6] | 23 |
| Obrázek 17. Design úrovní [6] | 23 |
| Obrázek 18. Trackview [6] | 24 |
| Obrázek 19. Flowgraph [6]..... | 25 |
| Obrázek 20. Integrace látek [6]..... | 25 |
| Obrázek 21. ACE [8] | 28 |
| Obrázek 22. Abstrakce zvuku [8] | 28 |
| Obrázek 23. Příklad zvukové okluze [8] | 29 |
| Obrázek 24. Dynamický systém odezvy [8]..... | 30 |
| Obrázek 25. Profilování [10] | 31 |
| Obrázek 26. Stascope [10]..... | 32 |
| Obrázek 27. Titulní strana dokumentace | 38 |
| Obrázek 28. Nový level | 40 |
| Obrázek 29. Pohled na upravený level z hráčovy perspektivy | 41 |
| Obrázek 30. Přidána vegetace do levelu | 43 |
| Obrázek 31. Pohled na skalní ohraničení levelu | 44 |
| Obrázek 32. Nastavení emitoru pro objekt Down | 45 |

| | |
|---|----|
| Obrázek 33. Nastavení emitoru pro objekt Top..... | 45 |
| Obrázek 34. Výsledná podoba vodopádu | 46 |
| Obrázek 35. Výsledná podoba řeky | 47 |
| Obrázek 36. Výsledná podoba cesty | 48 |
| Obrázek 37. Výsledná podoba vesnice | 49 |
| Obrázek 38. Výsledná podoba objektu cíle | 50 |
| Obrázek 39. Výsledná startovní lokace | 52 |
| Obrázek 40. Vložení příkazu | 52 |
| Obrázek 41. Nastavení prostředí..... | 53 |
| Obrázek 42. Výsledný pohled na oblohu..... | 53 |
| Obrázek 43. Výsledný efekt deště | 54 |
| Obrázek 44. Editace gamesdk.cryproject | 55 |
| Obrázek 45. Vytvořené úkoly v xml souboru..... | 56 |
| Obrázek 46. Vytvořené proměnné pro stavy úkolů | 56 |
| Obrázek 47. Inicializace proměnných při načtení levelu..... | 58 |
| Obrázek 48. Implementace prvního úkolu..... | 59 |
| Obrázek 49. Implementace druhého úkolu | 59 |
| Obrázek 50. Implementace třetího úkolu..... | 60 |
| Obrázek 51. Implementace čtvrtého úkolu | 60 |
| Obrázek 52. Implementace zobrazení aktuálního úkolu..... | 61 |
| Obrázek 53. Implementace upozornění | 61 |
| Obrázek 54. Výsledný efekt Lens Flare..... | 62 |
| Obrázek 55. Implementace logiky pro sledování cesty | 63 |
| Obrázek 56. Odstranění modelu nepřítele | 64 |
| Obrázek 57. Doplnění funkcionality navigace k cíli první mise | 65 |
| Obrázek 58. Opětovné načtení údajů aktuální mise po pozastavení hry | 65 |
| Obrázek 59. Deaktivace in-game debug informací | 66 |
| Obrázek 60. Hlavní menu | 66 |

SEZNAM TABULEK

| | |
|--|----|
| Tabulka 1 Unity vs CryEngine [12]..... | 35 |
| Tabulka 2 Unreal Engine vs CryEngine [13]..... | 36 |

SEZNAM PŘÍLOH

Příloha P I: USB flash disk obsahuje:

- Složka Project obsahující celý CryEngine projekt – spustitelný soubor **gamesdk.cryproject**
- Složka Visual C++ Redistributable for Visual Studio 2015-2022 obsahující instalátor VC_redist.x64.exe
- Soubor ve formátu PDF/A r_marecek_dokumentace_cryengine.pdf
- Složka SWIGame obsahující výslednou aplikaci – spustitelný soubor **SWIGame.exe** v adresáři **SWIGame\bin\win_x64**

Příloha P II: Snímky ze hry:





