Stredná priemyselná škola elektrotechnická

Hálova 16, 851 01 Bratislava

**Dwarven Adventures – počítačová hra**

KOMPLEXNÁ ODBORNÁ MATURITNÁ PRÁCA

Bratislava, 2025 4.D Adrián Peško

Stredná priemyselná škola elektrotechnická

Hálova 16, 851 01 Bratislava

**Dwarven Adventures – počítačová hra**

KOMPLEXNÁ ODBORNÁ MATURITNÁ PRÁCA

**Študijný odbor**: 2573M programovanie digitálnych technológií  
**Konzultant**: Dominik Zatkalík

Bratislava, 2025 4.D Adrián Peško

**Čestné vyhlásenie**

Vyhlasujem, že prácu stredoškolskej odbornej činnosti na tému Dwarven Adventures – počítačová hra, som vypracoval samostatne, s použitím uvedených literárnych zdrojov. Prácu som neprihlásil a ani neprezentoval v žiadnej inej súťaži, ktorá je pod gestorstvom MŠVVaM SR. Som si vedomý dôsledkov, ak uvedené údaje nie sú pravdivé.

.........................................................

V Bratislave, 20.2 2025 Adrián Peško

Abstrakt

Kľúčové slová :

Abstract

Keywords :

**Obsah**

[ÚVOD 6](#_Toc190330488)

[1 Použité Technológie 7](#_Toc190330489)

[1.1 Blender 7](#_Toc190330490)

[1.2 Fl Studio 8](#_Toc190330491)

[1.3 Unreal Engine 5 9](#_Toc190330492)

[1.4 Adobe substance Painter 10](#_Toc190330493)

[1.5 Git hub 10](#_Toc190330494)

[1.6 Porovnanie s inými technológiami 11](#_Toc190330495)

[2 Modelovanie v Blendery 13](#_Toc190330496)

[2.1 Modelovacie prostredie 13](#_Toc190330497)

[2.2 Modifikátory 14](#_Toc190330498)

[2.3 Tvorenie objektov 15](#_Toc190330499)

[2.4 Tvorenie Uv mapy 17](#_Toc190330500)

[2.5 Textúrovanie 18](#_Toc190330501)

[2.5.1 Naimportovanie modelu 18](#_Toc190330502)

[2.5.2 Pridanie základných materiálov 19](#_Toc190330503)

[2.5.3 Práca s maskami 19](#_Toc190330504)

[2.5.4 s Maľovanie textúr 20](#_Toc190330505)

[2.5.5 Proces exportu textúr 20](#_Toc190330506)

[2.6 Tvorenie postáv 20](#_Toc190330507)

[2.6.1 Sculpting Postáv 21](#_Toc190330508)

[2.6.2 Textúrovanie Postáv 21](#_Toc190330509)

[3 Animácie 21](#_Toc190330510)

[3.1 Príprava modelov na animovanie 22](#_Toc190330511)

[3.1.1 Topológia 22](#_Toc190330512)

[3.1.2 Rigovanie 23](#_Toc190330513)

[3.1.3 Weight paint 23](#_Toc190330514)

[3.1.4 Testovanie pohybu 24](#_Toc190330515)

[3.2 Tvorba animácií 24](#_Toc190330516)

[3.3 Keyframe animácia 24](#_Toc190330517)

[3.4 Inverse Kinematics (IK) 24](#_Toc190330518)

[3.5 Looping animácií 25](#_Toc190330519)

[3.6 Facial animácia (mimika) 25](#_Toc190330520)

[4 Herné mechaniky 26](#_Toc190330521)

[4.1 Herné mechaniky hráča 26](#_Toc190330522)

[5 Používateľské Rozhranie (UI) 27](#_Toc190330523)

[5.1 Ui 27](#_Toc190330524)

[5.2 Zvuky 27](#_Toc190330525)

[6 Marketingová stratégia 27](#_Toc190330526)

[6.1 marketingový plán 27](#_Toc190330527)

[7 Zoznam použitej literatúry 28](#_Toc190330528)

**Zoznam skratiek, značiek a symbolov**

ÚVOD

Videohry sú jednou z najobľúbenejších foriem zábavy dnešnej doby. Spojujú kreativitu, technické zručnosti a detailnú prácu, čo ma vždy fascinovalo. Preto som sa rozhodol venovať svoju maturitnú prácu tvorbe vlastnej počítačovej hry. Chcem nielen zabaviť hráča, ale zároveň si vyskúšať, čo všetko patrí k vývoju hry – od programovania, cez 3D dizajn až po riešenie rôznych problémov, ktoré pri tvorbe vznikajú.

Túto tému som si vybral preto, že hry sú mojou vášňou a vždy som chcel vedieť, ako presne fungujú ich mechaniky, grafika a interakcia s hráčom. Táto práca mi dáva možnosť využiť svoje znalosti z programovania a 3D modelovania v praxi a zároveň vytvoriť niečo, na čo môžem byť hrdý.

Mojím cieľom bolo podrobne popísať, ako vlastne vzniká hra, porovnať rôzne herné technológie a zostaviť vývojový plán. Hru som sa rozhodol vytvoriť v Unreal Engine 5, kde som naprogramoval základné mechaniky hráča a vytvoril jeho animácie. Okrem toho som sa venoval aj návrhu herného prostredia, zvukov a používateľského rozhrania (UI), ktoré zabezpečuje, aby sa hráč v hre dobre orientoval.

Na záver som sa pozrel aj na marketing a stratégiu propagácie, pretože dobrá hra potrebuje nielen kvalitný obsah, ale aj správny spôsob, ako ju dostať k ľuďom. Táto práca mi dala veľa nových skúseností. Naučil som sa lepšie plánovať, riešiť problémy a hlavne skombinovať kreativitu s logickým myslením pri tvorbe herných mechaník.

1. Použité Technológie

Pri tvorbe tohto projektu sme využívali rôzne technológie a nástroje, ktoré nám umožnili realizovať projekt od návrhu až po finálnu verziu. Ako hlavný vývojový nástroj sme zvolili Unreal Engine 5, ktorý nám poskytol široké možnosti pre tvorbu 3D hry a zároveň intuitívne rozhranie na prácu s hernými objektmi a mechanikami.

Programovaciu stránku sme realizovali pomocou blueprintov, ktoré sú súčasťou Unrealu. Tieto blueprinty, bežné v hernom priemysle, nám umožnili jednoducho implementovať komplexné mechanizmy.

Pri tvorbe grafickej časti našej hry sme pracovali s nástrojom Blender, kde sme vytvárali vlastné postavy a prostredie. Na tvorbu textúr sme použili Adobe Substance Painter. Zvukové efekty a hudbu sme vytvárali a upravovali pomocou FL Studia.

Verziu pre testovanie sme spravovali cez GitHub, čo nám umožnilo prácu z rôznych počítačov. Tieto technológie nám poskytli potrebnú flexibilitu a podporu, aby sme dokázali realizovať naše predstavy a vytvoriť funkčný herný projekt.

* 1. Blender

[1](Blender foundation 2019, Editors) Blender je nástroj na tvorbu 3D modelov, animácií a vizualizácií, ktorý patrí medzi najobľúbenejšie softvéry v oblasti digitálneho umenia. Jeho všestrannosť a dostupnosť ho robia ideálnym nielen pre profesionálov, ale aj pre začiatočníkov. Blender nám umožňuje vytvárať jednoduché aj komplexné 3D objekty, textúry, osvetlenie a fyzikálne simulácie, ktoré môžeme využiť v hrách, filmoch alebo vizuálnych efektoch.

**Medzi jeho kľúčové funkcie patria:**

* **3D modelovanie:** Tvorba modelov pomocou rôznych nástrojov, ktoré nám umožňujú detailnú prácu s geometriou.
* **Animácie:** Možnosť vytvárať realistické animácie postáv, pohybov a objektov, čo je ideálne pre filmovú alebo hernú tvorbu.
* **Rendering:** Blender obsahuje dva výkonné enginy – Cycles a Eevee – ktoré ponúkajú realistické alebo rýchle renderovanie podľa našich potrieb.
* **Simulácie:** Podpora pre simulácie fyziky, ako sú tekutiny, oheň, dym alebo vlasy, čo nám umožňuje realistické zobrazenie rôznych efektov.
* **Integrácia s inými nástrojmi:** Podpora importu a exportu rôznych formátov (napr. OBJ, FBX), čo nám umožňuje používať vytvorené modely v herných enginoch, ako sú Unity alebo Unreal Engine.

Pri tvorbe nášho projektu sme sa sústredili hlavne na 3D modelovanie a animácie. Blender nám poskytol dostatok možností na vytvorenie profesionálnych modelov a ich oživenie prostredníctvom prepracovaných animácií.

* 1. Fl Studio

[2](Image-Line 2025, What can FL Studio do?) FL Studio, známe aj ako Fruity Loops, je softvér na tvorbu a úpravu hudby, ktorý si získal popularitu medzi hudobnými producentmi, skladateľmi a kreatívcami. Tento nástroj nám poskytuje širokú škálu funkcií na tvorbu hudby, ale aj rôznych zvukových prvkov, ktoré možno využiť napríklad v hrách – od autentických zvukov prostredia až po podkladovú hudbu, ktorá umocňuje atmosféru herného prostredia.

**Kľúčové funkcie FL Studia zahŕňajú:**

* **Piano Roll:** Prepracovaný a intuitívny nástroj na komponovanie melódií, akordov a basových liniek, ktorý nám umožňuje detailnú prácu so zvukmi.
* **Mixér:** Výkonné možnosti mixovania a masteringu, ktoré nám dávajú priestor na pridávanie efektov a úpravu zvuku na profesionálnej úrovni.
* **VST Pluginy:** Podpora širokého spektra pluginov, ktoré rozširujú zvukové možnosti, ako sú syntetizátory, efekty a virtuálne nástroje.
* **Step Sequencer:** Praktický nástroj na tvorbu beatov a rytmických sekvencií, ktorý zjednodušuje našu prácu s rytmickými prvkami.
* **Široké spektrum zvukov:** K dispozícii máme množstvo prednastavených nástrojov, samplov a efektov, ktoré nám umožňujú kreatívnu a efektívnu prácu.
* **Automatizácia:** Funkcia, ktorá nám umožňuje v reálnom čase riadiť parametre nástrojov a efektov, čím dosahujeme dynamickejšie a komplexnejšie zvukové výsledky.

Z nášho pohľadu patrí medzi najpoužívanejšie funkcie mixér, ktorý sme využili na úpravu samplov a docielili tak požadovanú kvalitu zvuku. FL Studio nám zároveň poskytlo veľkú mieru flexibility, vďaka čomu sme mohli experimentovať a vytvárať zvuky presne podľa našich predstáv.

* 1. Unreal Engine 5

[4](Epic Games 2024, Tools and Editors) Unreal Engine je výkonný herný engine vyvinutý spoločnosťou Epic Games, ktorý sa stal jedným z najpopulárnejších nástrojov na tvorbu hier, vizualizácií a interaktívnych aplikácií. Tento engine je známy svojou schopnosťou generovať vysoko kvalitné 3D grafiky, realistické prostredia a komplexné herné mechaniky. Používa sa nielen v indie projektoch, ale aj v AAA hrách, čo ukazuje jeho širokú použiteľnosť a silu.

**Medzi kľúčové funkcie Unreal Engine patria:**

* **Vysokokvalitná grafika:** Unreal Engine je známy pokročilými vizuálnymi efektmi, realistickým osvetlením a podporou najnovších technológií, ako je ray tracing, ktorý umožňuje tvorbu fotorealistických prostredí.
* **Blueprints Visual Scripting:** Tento vizuálny skriptovací systém poskytuje tvorcom hier nástroj na vytváranie hernej logiky bez potreby písania kódu, čo výrazne urýchľuje vývoj.
* **C++ Programovanie:** Pre pokročilých používateľov Unreal Engine ponúka flexibilitu a optimalizáciu prostredníctvom priamych úprav hernej logiky v jazyku C++, čo umožňuje neobmedzené prispôsobenie.
* **Kreatívne nástroje:** Engine obsahuje širokú paletu nástrojov na tvorbu 3D modelov, animácií, textúr a zvukov, čím zjednodušuje celý proces vývoja priamo v prostredí enginu.
* **Podpora pre VR a AR:** Unreal Engine poskytuje silnú podporu pre vývoj hier a aplikácií určených pre virtuálnu realitu (VR) a rozšírenú realitu (AR), čím umožňuje vytvárať pokročilé interaktívne zážitky.
* **Real-time Rendering:** Unreal Engine je známy svojím výkonom v oblasti renderovania v reálnom čase, čo je ideálne pre tvorbu hier, ale aj pre vizualizácie a simulácie v reálnom čase, ktoré sú potrebné pre moderné aplikácie.

Tento engine nám poskytol flexibilitu a nástroje, ktoré sme potrebovali na vytvorenie náročných herných mechaník, kvalitnej grafiky a efektívneho vývoja projektu.

* 1. Adobe substance Painter

[3](Adobe 2021, ASM specifications) Substance Painter je profesionálny nástroj na textúrovanie a maľovanie 3D modelov, ktorý sa stal štandardom v hernom a filmovom priemysle. Vyvinutý spoločnosťou Adobe, Substance Painter nám umožňuje vytvárať vysoko detailné a realistické textúry na 3D objektoch v reálnom čase. Tento nástroj je obzvlášť obľúbený pre svoju flexibilitu a schopnosť vytvárať pokročilé textúry bez nutnosti prechádzať zložité procesy, čo výrazne zjednodušuje tvorbu komplexných povrchových detailov.

Medzi kľúčové funkcie Substance Painter patria:

nástrojmi, ako je Adobe Photoshop, čo nám umožňuje ľahkú úpravu textúr a pridávanie ďalších detailov.

Substance Painter nám poskytol širokú paletu nástrojov na textúrovanie modelov, pričom sme ho využili aj na baking high poly textúr na low poly mesh, čím sme dosiahli požadovaný detail a realizmus pri zachovaní optimálnych výkonových parametrov.

* 1. Git hub

[5](GitHub, Inc. 2025, About GitHub and Git) GitHub je platforma na správu a verzionovanie kódu, postavená na systéme Git, ktorá sa stala nevyhnutným nástrojom pre spoluprácu medzi vývojármi. Poskytuje efektívne sledovanie zmien v kóde, možnosť pracovať na rôznych verziách projektu a spravovať tímovú spoluprácu pri vývoji softvérových aplikácií. Tento nástroj je obzvlášť cenný pre projektový manažment a vývojárov, ktorí potrebujú efektívne organizovať a udržiavať projekty.

**Kľúčové funkcie GitHubu zahŕňajú:**

* **Maľovanie v reálnom čase:** Umožňuje nám natierať textúry priamo na 3D modely, pričom všetky zmeny sú okamžite zobrazené, čo zjednodušuje prácu a urýchľuje tvorbu textúr.
* **Pokročilé vrstvy a masky:** Nástroj podporuje vrstvy, masky a štýly, čo nám umožňuje detailné úpravy a experimentovanie s rôznymi efektmi, ako sú poškriabania, opotrebovanie alebo špina, ktoré pridávajú hĺbku a realizmus modelom.
* **PBR Workflow (Physically-Based Rendering):** Substance Painter je navrhnutý pre moderný PBR workflow, čo znamená, že textúry vytvorené v ňom vyzerajú realisticky vo väčšine herných enginov, ako sú Unreal Engine alebo Unity, a lepšie sa prispôsobujú fyzikálnym vlastnostiam materiálov.
* **Vytváranie dynamických efektov:** Tento nástroj umožňuje vytvárať rôzne materiály a efekty, ako kov, drevo, kožu, tkaniny alebo zoxidované povrchy, ktoré sa môžu zobrazovať v reálnom čase pri práci s modelmi.
* **Import modelov a exportovanie textúr:** Jednoduchá práca s 3D modelmi vo formátoch ako OBJ alebo FBX, pričom hotové textúry môžeme exportovať do formátov, ktoré sú kompatibilné s hernými enginmi a ďalšími grafickými nástrojmi.
* **Interakcia s inými nástrojmi Adobe:** Substance Painter sa hladko integruje s ďalšími programami.
* **GitHub Pages:** S GitHubom môžeme vytvárať a hostovať statické webové stránky priamo z repozitára, čo je ideálne pre prezentácie projektov alebo dokumentáciu.

GitHub sme použili ako uložisko pre náš projekt, kde sme spravovali verzie kódu a uľahčili tímovú spoluprácu.

* 1. Porovnanie s inými technológiami

Pri výbere nástrojov pre náš projekt sme sa rozhodli pre technológie, ktoré najlepšie vyhovovali našim potrebám a poskytovali optimálne výsledky. Tvorba grafiky nebola náročná na rozhodovanie, pretože s Blenderom už máme skúsenosti a tento nástroj je veľmi všestranný. Blender je ideálny na modelovanie, textúrovanie, animácie a ďalšie grafické úpravy. V porovnaní s inými komerčnými nástrojmi, ako Maya, je Blender tiež výhodný, pretože je zadarmo, čo je oproti drahým softvérom veľká výhoda. Maya je síce silný nástroj, ale je viac orientovaný na tvorbu postáv a animácie, čo sme v tomto projekte nevyužívali.

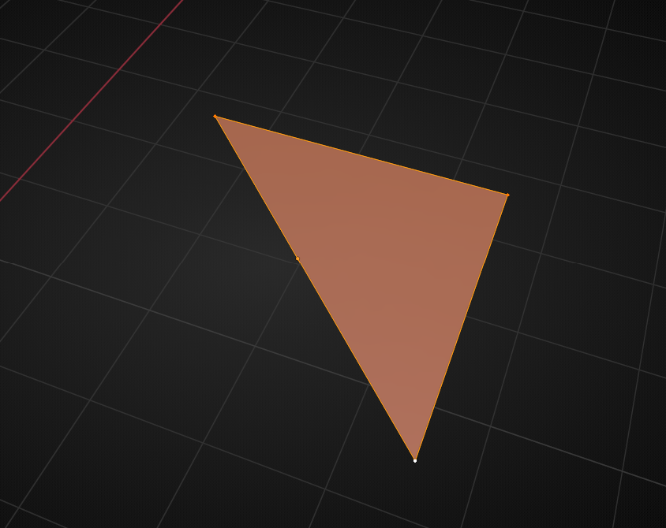
* **Verzionovanie kódu**: GitHub nám umožňuje sledovať každú zmenu v kóde a vrátiť sa k predchádzajúcim verziám, čím sa minimalizuje riziko straty práce a uľahčuje správa zmien v projekte.
* **Spolupráca**: Umožňuje nám pracovať na rôznych vetvách (branches) projektu a následne ich spojiť (merge), čo zjednodušuje tímovú prácu a integráciu nových funkcií.
* **Pull Requests**: GitHub podporuje proces navrhovania zmien prostredníctvom pull requests, ktoré nám umožňujú diskutovať a schvaľovať zmeny pred ich implementovaním do hlavnej vetvy kódu.
* **Issue Tracking**: GitHub poskytuje nástroj na sledovanie chýb, úloh a požiadaviek, čo nám pomáha organizovať a efektívne manažovať projekty.
* **Integrácie a CI/CD**: GitHub podporuje integráciu s nástrojmi na nepretržité testovanie a nasadzovanie (CI/CD), čo automatizuje proces testovania a nasadzovania kódu.

Pri výbere medzi Unreal Engine a Unity sme zvážili niekoľko kľúčových faktorov. Obidve platformy sú veľmi silné a populárne pre tvorbu hier, ale každá má svoje špecifiká. Pre náš projekt sme sa rozhodli pre Unreal Engine, pretože ponúka množstvo nástrojov a knižníc, ktoré uľahčujú vývoj, a jeho vizuálny skriptovací systém Blueprints sa nám zdá jednoduchší a intuitívnejší. Tento nástroj nám umožnil rýchlo implementovať hernú logiku bez potreby písania kódu, čo výrazne zjednodušilo vývoj a ušetrilo čas.

Hoci Unity je flexibilné a obľúbené hlavne medzi menšími vývojármi, Unreal Engine nám poskytol všetky nástroje, ktoré sme potrebovali, ako pokročilé vizuálne efekty a realistické možnosti renderovania. Preto sme sa rozhodli pre Unreal Engine, ktorý nám umožnil efektívne vytvoriť hru s kvalitným vizuálom a rozsiahlou podporou herných mechanizmov.

1. Modelovanie v Blendery

(Blender foundation 2019, Mesh Operators) Face objektu v Blenderi je základná stavebná jednotka 3D geometrie, ktorá predstavuje plochu uzatvorenú medzi **edges** (hranami) a **vertices** (vrcholmi). Tieto plochy spolu vytvárajú povrch objektu a umožňujú jeho zobrazenie, úpravu a textúrovanie. Každý face je súčasťou **mesh** (sieťovej štruktúry) modelu a je kľúčový pri vytváraní realistických 3D modelov, pretože práve cez neho sa definujú tvary a detaily objektu. Na obrázku 1 vidíme trojuholník ktorý sa skladá z 3 vrcholov, 3 hrán a jednej strany, ktoré spolu tvoria polygón (trojuholník).



Obr. 1 Face objektu v Blendery (Ukážka hlavných jednotiek 3D geometrie)

* 1. Modelovacie prostredie

(Blender foundation 2019, User Interface) Blender vnímame ako 3D modelovacie prostredie, ktoré sa vyznačuje elegantným a prehľadným rozhraním. Po jeho spustení sa ocitneme na **hlavnej pracovnej ploche**, ktorá je rozdelená na niekoľko kľúčových oblastí:

* **3D Viewport**: Najväčšia časť obrazovky, kde môžeme vidieť a pracovať s 3D modelmi. Nachádza sa tu **mriežka** na orientáciu, 3D kurzor a jednoduché objekty, ako je predvolená kocka.
* **Outliner**: Na pravej strane obrazovky objavujeme okno so zoznamom všetkých objektov v scéne, ktoré nám umožňuje efektívne vyhľadávať a organizovať scény.
* **Properties Panel**: V spodnej pravej časti nachádzame panel so záložkami, kde spravujeme vlastnosti objektov, ako sú materiály, modifikátory, svetlá či renderovacie nastavenia.
* **Timeline**: Pri práci na animáciách nás zaujíma časová os v dolnej časti, kde môžeme upravovať kľúčové snímky a plynulosť animácií.
* **Toolbar**: Na ľavej strane obrazovky vidíme panel s nástrojmi, ktoré využívame na modelovanie, ako napríklad **extrúziu**, **rezanie** či **zrkadlenie**.

Obrázok, na ktorom je text, snímka obrazovky, multimediálny softvér, softvér

Automaticky generovaný popisCelé prostredie si môžeme prispôsobiť – okná sa dajú presúvať, meniť ich veľkosť alebo dokonca vytvárať úplne nové rozloženia. Blender pracuje s príjemným **tmavým motívom** doplneným o farebné ikony, čo nám uľahčuje sústredenie pri dlhodobej práci. Navigácia v 3D priestore je intuitívna – využívame stredné tlačidlo myši na otáčanie, **Shift** na posúvanie a **scrollovanie** na približovanie. Na obrázku číslo 2 môžeme vidieť umiestnenie hlavných UI elementov programu Blender.

Obr. 2 3D ViewPort (Ukážka UI elementov programu Blender)

* 1. Modifikátory

(Blender foundation 2019, Mesh Operators) Modifikátory v Blenderi sú silné nástroje, ktoré nám umožňujú vykonávať rôzne úpravy na 3D objektoch bez toho, aby sme ich priamo a nenávratne zmenili. Nachádzajú sa v Properties Paneli pod ikonou kľúča (Modifiers tab) a ich aplikácia je jednoduchá, no zároveň otvára veľa možnosti pri tvorbe a úprave modelov.

Každý modifikátor nám poskytuje špecifickú funkciu. Napríklad pomocou Subdivision Surface modifikátora môžeme objekty zjemniť a rozdeliť ich na viac detailných častí, čo je ideálne pri tvorbe organických tvarov a pri Sculptingu (tvarovanie). Array modifikátor nám umožňuje duplikovať objekty v radoch alebo stĺpcoch, pričom vieme presne kontrolovať ich rozmiestnenie. Mirror modifikátor zasa využívame na symetrické modelovanie, čo nám výrazne šetrí čas pri práci na zrkadlovo rovnakých častiach objektov.

Niektoré modifikátory, ako napríklad Boolean, nám umožňujú kombinovať alebo odčítať objemy objektov, čo nám otvára nové možnosti pri tvorbe komplexných tvarov. Ďalšie, ako Bevel, pridávajú detail v podobe zaoblených hrán, ktoré pôsobia realistickejšie. Pri práci s deformáciami môžeme siahnuť po modifikátoroch ako Displace alebo Wave, ktoré nám umožnia meniť tvary objektov pomocou textúr alebo simulácií.

Obrázok, na ktorom je snímka obrazovky, 3d modelovanie, grafický softvér, text

Automaticky generovaný popisDôležité je, že modifikátory môžeme vrstviť, čo nám dáva možnosť vytvárať komplikované efekty postupne a kontrolovane. Kým sú aplikované ako modifikátory, zostávajú tieto úpravy nedeštruktívne, čo znamená, že sa k pôvodnému tvaru objektu môžeme vždy vrátiť. Ak však potrebujeme úpravy „zafixovať“, môžeme modifikátor aplikovať (Apply), čím sa stane trvalou súčasťou objektu. Na obrázku číslo 3 môžeme vidieť kocku, ktorá bola upravená pomocou modifikátorov.

Obr. 3 Modifikátory (Modifikácia objektu pomocou Bevel a Boolean)

* 1. Tvorenie objektov

Predtým ako začneme modelovať objekty, je potrebné si najprv premyslieť, čo budeme potrebovať v našej hre, a rozhodnúť sa pre vizuálny štýl alebo obdobie, v ktorom bude hra zasadená. V našom prípade sa naša hra odohráva v stredoveku. Po stanovení štýlu a obdobia môžeme pokračovať s tvorbou zoznamu objektov, ktoré budeme potrebovať.

V našom projekte sme identifikovali tri hlavé kategórie objektov: environmentálne objekty (objekty prostredia), interaktívne objekty a hrateľné objekty. Po tejto fáze nasleduje hľadanie inšpírácií, ktoré sa hodia do našej hry. Je dôležité dbať na to, aby sme dodržiavali vopred stanovený štýl, čím zabezpečíme vizuálnu konzistenciu.

Obrázok, na ktorom je symbol

Automaticky generovaný popisKeď máme dostatok inšpiračných materiálov, prechádzame na samotné modelovanie. Začíname s prvotným hrubým nákresom modelu, ktorý je znázornený na obrázku číslo 4.

Obr. 4 Jednoduchý model (Ukážka hrubého nákresu modelu)

Tento nákres slúži ako základ, na ktorý neskôr aplikujeme rôzne úpravy. Pri modelovaní sa zameriavame na optimalizáciu geometrie, čo zahťňa úpravy, ako je spájanie vertexov (merging vertices), čím zabezpečíme, že sa na rovnakom mieste nenachádzajú zbytočné vertexi. Okrem toho využívame modifikátor Bevel na zaoblenie hrán a aplikujeme funkciu Shade Auto Smooth, aby sme dosiahli hladšie povrchy.

Ak chceme na objekte vytvárať špecifické detaily, ako napríklad výrezy, používame nástroj Boolean. Výsledok tohto postupu je znázornený na obrázku číslo 5. Po dokončení a doladení modelu prechádzame na jeho prípravu na textúrovanie. Tento krok zahŕňa zabezpečenie, že model spĺňa technické požiadavky pre aplikáciu textúr a vytváranie optimálnych UV máp.



Obr. 5 Jednoduchý model (Ukážka upraveného modelu)

* 1. Tvorenie Uv mapy

Tvorba UV máp (rozloženie 3D modelu na 2D plochu pre aplikáciu textúr, zvýraznené na obrázku číslo 6 naľavo) je jedným z najdôležitejších krokov pri príprave modelu na textúrovanie. Začíname označením seamov, čo sú hranice, podľa ktorých sa model „rozloží“. Seam (obr. č. 6 napravo) označujeme na miestach, kde budú najmenej viditeľné – napríklad na spodnej strane objektov alebo v menej exponovaných častiach modelu.

Keď máme seamy označené, použijeme funkciu Unwrap (rozbalenie), ktorá automaticky rozloží povrch modelu do UV mapy. Ak je výsledná mapa nevyhovujúca, manuálne ju upravujeme. Snažíme sa zabezpečiť, aby textúry neboli naťahované ani deformované. Na kontrolu využívame špeciálne textúry, ktoré nám pomôžu identifikovať problémy v UV mape.

Na jednoduchšie objekty môžeme využiť nástroj Smart UV Project, ktorý automaticky vygeneruje UV mapu. Tento nástroj je užitočný pri menej komplikovaných modeloch, ale pri zložitejších je často potrebné manuálne nastavenie seamov a precízna úprava UV mapy.

Obrázok, na ktorom je text, snímka obrazovky, grafický softvér, multimediálny softvér

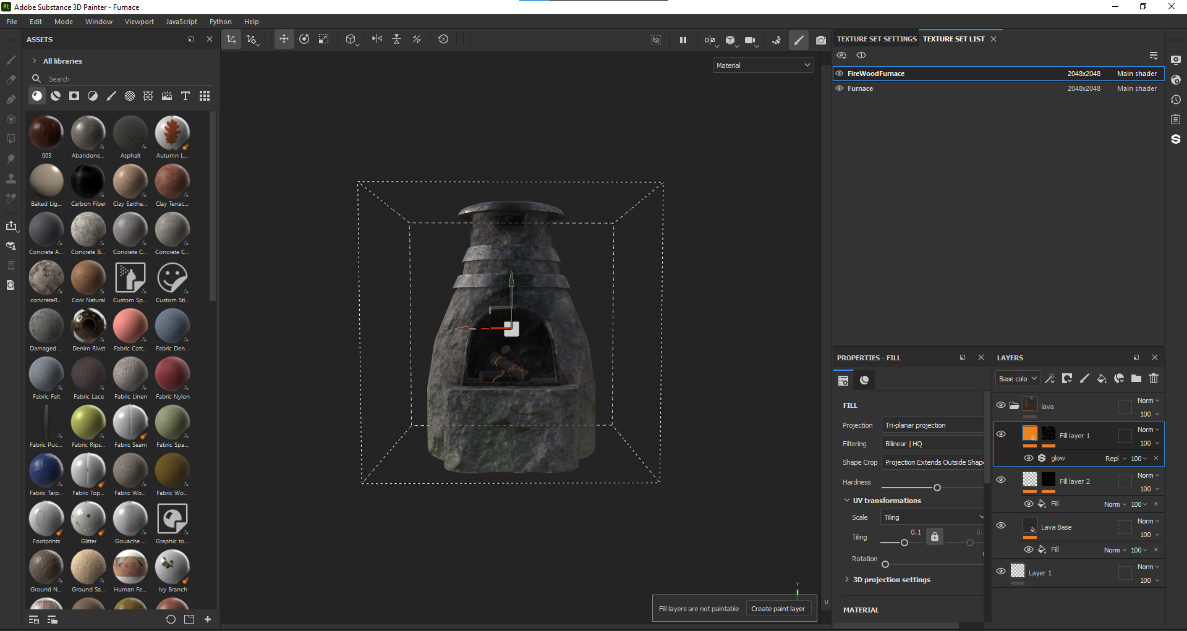
Automaticky generovaný popisPo dokončení UV mapovania dôkladne kontrolujeme kvalitu rozloženia. V textúrovacom režime overujeme, či sú textúry správne aplikované na všetky časti modelu. Tento krok je obzvlášť dôležitý, pretože kvalita UV máp priamo ovplyvňuje, ako realisticky a profesionálne bude model vyzerať v hre. Po ukončení tohto kroku môžeme pokračovať s exportom fbx súboru modelu.

Obr. 6 Model s hotovou UV mapou

* 1. Textúrovanie
     1. Naimportovanie modelu

Na začiatok si do Substance Paintera naimportujeme hotový 3D model s UV mapou. Pri vytváraní nového projektu si vyberieme šablónu pre engine, ktorý používame v našom prípade Unreal Engine, aby sme mali správne nastavené všetky parametre pre textúry. Potom nastavíme rozlíšenie textúr – najčastejšie 2K alebo 4K, podľa toho, akú kvalitu chceme dosiahnuť. Pred samotným textúrovaním je dôležité spraviť aj baking, čo je vlastne proces, kde sa prepočítajú rôzne detaily modelu, ako sú tiene, normálové mapy, svetelné informácie alebo rôzne povrchové detaily, a tieto sa „bake-ujú“ do statických textúr. Baking nám uľahčuje prácu, pretože sa už nemusíme starať o svetlo a tiene pri nanášaní textúr – všetko je už zahrnuté v textúrach. Týmto spôsobom si ušetríme čas a získame lepší vizuálny efekt, pretože sa tieto detaily prejavia na modeli oveľa realistickejšie a šetrnejšie k výkonom hry.

* + 1. Pridanie základných materiálov

Keď máme model pripravený, začneme s aplikovaním základných materiálov. V Substance Painteri nájdeme obrovskú knižnicu rôznych materiálov, ako sú kov, drevo, plast a mnoho ďalších (vieme ich nájsť na obrázku číslo 7 naľavo). Tieto materiály ďalej rozdeľujeme na base materiály a smart, najvýraznejší rozdiel medzi nimi je v množstve detailov a ich prevedení, kde pri base materiáloch poloha detailov je vopred určená, smart materiály ich odvodzujú od tvaru modelu a máp ktoré sa vytvorili počas bakingu. Tieto materiály ďalej môžeme jednoducho naniesť na náš model a prispôsobiť ich podľa potreby. Ak nám nevyhovuje základný vzhľad, každý materiál je možné upraviť – môžeme zmeniť farbu, textúru, drsnosť povrchu, lesk alebo dokonca priesvitnosť. Týmto spôsobom máme úplnú kontrolu nad tým, ako bude každý povrch vyzerať, a môžeme model prispôsobiť tak, aby vyzeral presne podľa našich predstáv. Tieto úpravy sa uskutočňujú v properties tabe. Ďalšie taby sú napríklad layers a texture set list.

Obr. 7 Textúrovanie (Ukážka UI programu Adobe Substace Painter)

* + 1. Práca s maskami

Masky sú silný nástroj, ktorý nám umožňuje veľmi precízne určiť, kde sa jednotlivé materiály aplikujú na model (jednotlivé masky a materiály sa nachádzajú na obrázku číslo 7 napravo dole). Napríklad, ak chceme na kovovom povrchu vytvoriť efekt škrabancov alebo opotrebovania, využijeme „smart masky“. Tieto masky automaticky pridajú detaily ako zodraté hrany, nečistoty v rohoch alebo špinu na povrchu, čo z modelu robí oveľa realistickejší objekt. Masky tohto typu vieme nájsť v okne ktoré sa nachádza naľavo pod záložkou Smart masks. Masky môžeme tiež upravovať a kombinovať rôznymi spôsobmi, aby sme dosiahli ešte detailnejší efekt, či už ide o rôzne vrstvy poškodenia alebo rozdiely v textúre.

* + 1. s Maľovanie textúr

Jedným z najzábavnejších krokov v celom procese je maľovanie textúr priamo na model. V Substance Painteri máme k dispozícii širokú paletu štetcov a nástrojov, ktoré nám umožňujú vytvárať veľmi presné a detailné textúry. Ak potrebujeme pridávať jemné detaily ako drevené vlákna, štruktúru kameňa, zaprášenie objektu alebo iné defekty na povrchu modelu, toto zaručí že objekt vyzerá realistickejšie a lepšie zapadne do prostredia. Pomocou štetcov môžeme model doslova „maľovať“ – od jemných odtieňov po zložité vzory, ktoré dodajú objektu hĺbku a textúru, akú potrebujeme. Brushes (štetce) sa taktiež nachádzajú v ľavom okne pod záložkou Brushes.

* + 1. Proces exportu textúr

Keď máme model kompletne natextúrovaný, ďalším krokom je exportovanie textúr, aby sme ich mohli použiť v hernom engine alebo inom softvéri. Najprv si otvoríme záložku „Export Textures“, kde môžeme prispôsobiť nastavenia exportu podľa našich potrieb.

V prvom rade si vyberieme exportnú šablónu (Template) z ponuky. Substance Painter ponúka prednastavené šablóny pre rôzne herné enginy, ako sú Unreal Engine, Unity, alebo štandardné PBR (Physically Based Rendering) šablóny. Tieto šablóny zabezpečia, že textúry budú exportované vo formáte a nastavení, ktoré sú kompatibilné s vybraným enginom. Ak pracujeme napríklad s Unreal Engine, šablóna automaticky zahrnie všetky potrebné mapy, ako sú Base Color, Roughness, Metallic, a Normal Map.

* 1. Tvorenie postáv

Vytvorenie vlastnej 3D postavy v Blendery môže na prvý pohľad vyzerať ako poriadna výzva, ale keď sa do toho dostaneme, zistíme, že ide o kreatívny proces, ktorý sa dá zvládnuť krok za krokom. Od základného modelovania cez textúrovanie až po prípravu na animáciu – každý z týchto krokov má svoje miesto a spolu tvoria kompletný workflow.

Na začiatku si vytvoríme základné tvary, aby sme mali predstavu o proporciách a siluete postavy. Tomuto kroku sa hovorí **blokovanie (blockout)** a slúži na rýchle nastavenie celkovej štruktúry modelu. Pracujeme tu s primitívnymi tvarmi, ako sú gule, kocky alebo valce, ktoré nám pomôžu určiť veľkosť hlavy, tela, rúk a nôh. Cieľom je čo najlepšie vystihnúť tvar postavy ešte predtým, než sa pustíme do detailov.

* + 1. Sculpting Postáv

Keď sme spokojní so základnou formou, môžeme prejsť na **modelovanie detailov**. Ak chceme dosiahnuť realistický vzhľad, použijeme **sculpting (sochárčenie)**, kde postupne pridávame svaly, záhyby kože alebo iné organické prvky. Tu si treba dať pozor na správnu anatómiu – môžeme použiť referenčné obrázky alebo anatomické modely, aby postava nepôsobila neprirodzene. Pri stylizovaných postavách máme viac slobody a môžeme experimentovať s proporciami alebo výraznejšími črtami.

Po sculptingu má model často obrovské množstvo polygónov, čo je problém hlavne v hrách a animáciách, kde potrebujeme optimalizovať výkon. Preto prichádza na rad **retopológia**, čo znamená, že si vytvoríme jednoduchšiu verziu modelu s čistou geometriou. Tento krok je dôležitý nielen kvôli výkonu, ale aj kvôli správnemu animovaniu – ak máme zle urobenú topológiu, model sa pri pohybe môže deformovať na nesprávnych miestach.

* + 1. Textúrovanie Postáv

Aby sme mohli na postavu naniesť realistické farby a textúry, musíme spraviť **UV mapovanie**. To znamená, že model doslova „rozbalíme“ do 2D priestoru, aby sme naň mohli aplikovať textúry bez deformácií. Po vytvorení UV mapy sa pustíme do **textúrovania**, kde môžeme ručne maľovať detaily v Blenderi alebo ich vytvoriť v špecializovaných softvéroch, ako je Substance Painter. V tomto kroku pridávame farbu pleti, očí, oblečenia či rôzne efekty, ako sú jazvy, tetovania alebo škrabance na brnení.

1. Animácie

Animovanie v Blendere je výkonný, ale pre začiatočníkov možno trochu komplikovanejší proces. Keď sa však oboznámime s jeho nástrojmi, môžeme vytvárať plynulé pohyby postáv, objektov alebo aj celých scén. Blender ponúka keyframe animáciu, čo znamená, že nastavujeme dôležité body pohybu v určitých časových bodoch a Blender medzi nimi automaticky doplní prechody.

Všetko sa deje v Dope Sheete a Graph Editore, kde si môžeme upravovať časovanie a plynulosť animácie. Ak animujeme postavu, využívame rigging, teda pridanie kostí (armature), ktoré pohybujú jednotlivými časťami modelu. Pre realistické pohyby môžeme použiť Inverse Kinematics (IK), čo znamená, že pohyb jednej časti automaticky ovplyvní ostatné – napríklad keď posunieme ruku, prirodzene sa prispôsobí aj rameno.

Okrem manuálnej animácie môžeme v Blenderi použiť aj motion capture dáta, kde si importujeme reálne pohyby a aplikujeme ich na postavu. Ak pracujeme s kamerou, môžeme ju animovať pomocou trackovania objektov alebo nastaviť rôzne efekty, ako je pohybujúce sa pozadie alebo dynamické svetlá.

* 1. Príprava modelov na animovanie

Aby sme mohli našu 3D postavu animovať, nestačí ju len pekne vymodelovať a otextúrovať. Musíme ju správne pripraviť tak, aby sa pri pohybe nedeformovala a aby animácia prebiehala hladko. Tento proces zahŕňa niekoľko kľúčových krokov: **správna topológia, rigging, váhovanie (weight painting) a testovanie pohybov**.

* + 1. Topológia

Prvým krokom je zabezpečiť, aby náš model mal **čistú a dobre optimalizovanú geometriu**. Ak má model zlú topológiu (napríklad chaotické rozloženie polygónov alebo trojuholníky na kĺboch), pri animácii sa bude deformovať neprirodzene. Preto používame **štvoruholníkovú (quad) topológiu**, ktorá umožňuje plynulé ohýbanie a správnu interpoláciu pohybov.

Najdôležitejšie oblasti, kde treba mať dobrú topológiu, sú:

* **Kĺby** – lakte, kolená, ramená a prsty by mali mať viacero hrán, aby sa ohýbali postupne a nie ako "zlomené" časti.
* **Tvár** – ak plánujeme robiť animáciu mimiky, musíme dbať na **smerovanie edge loopov**, aby sa pohyby pier, očí a obočia dali dobre animovať.
* **Plynulé prechody** – napríklad medzi trupom a rukami alebo krkom a hlavou, aby sa časti tela pohybovali realisticky.

Ak model obsahuje príliš veľa polygónov, môžeme urobiť **retopológiu** a zjednodušiť ho tak, aby bol vhodný na animáciu bez straty detailov.

* + 1. Rigovanie

Keď máme hotový model, potrebujeme doň pridať **rig**, teda kostru (Armature), ktorá umožní jeho pohyb. V Blenderi môžeme rigging robiť ručne alebo použiť **Auto-Rigging nástroje**, ako je **Rigify**, ktorý nám vytvorí predpripravený rig pre humanoidné postavy.

Proces riggingu prebieha takto:

1. **Vytvoríme Armature** – pridáme prvú kosť (**Shift + A → Armature**) a postupne vytvoríme hierarchiu kostí (hlava, krk, ruky, nohy…).
2. **Priradíme kosti k postave** – umiestnime kosti tak, aby kopírovali prirodzenú štruktúru tela.
3. **Použijeme Inverse Kinematics (IK)** – pre nohy a ruky pridáme IK riešenie, aby sme mohli končatiny ovládať jednoduchšie.
4. **Prepojíme kostru s modelom** – použijeme „Parent With Automatic Weights“ (**Ctrl + P** → With Automatic Weights), aby Blender automaticky rozdelil vplyv kostí na model.
   * 1. Weight paint

Po pripojení modelu k rig-u musíme upraviť **váhy (weights)**, aby Blender vedel, ktorá časť modelu sa má pohybovať s ktorou kosťou. Toto sa robí pomocou **Weight Paintingu** – farebného mapovania, ktoré určuje, ako veľmi jednotlivé časti modelu reagujú na pohyb kostí.

* **Červená farba** znamená maximálny vplyv kosti na danú časť modelu.
* **Modrá farba** znamená, že kosť nemá na danú časť žiadny vplyv.
* **Prechody medzi farbami** určujú plynulé deformácie.

Ak váhy nie sú správne nastavené, môžu sa pri animácii objaviť problémy – napríklad pri pohybe ruky sa môže hýbať aj časť trupu, čo vyzerá divne. Preto si vždy skontrolujeme pohyby a manuálne doladíme váhy, kde je to potrebné.

* + 1. Testovanie pohybu

Keď máme model zvážený, musíme ho otestovať. Skúsime pohýbať rukami, nohami, ohnúť chrbticu či otočiť hlavu a sledujeme, či sa všetko správa správne. Ak sa nejaká časť deformuje zvláštne, upravíme váhy alebo topológiu.

Na testovanie môžeme použiť aj **pose mode**, kde nastavíme rôzne pózy a sledujeme, ako sa postava správa. Ak všetko vyzerá v poriadku, môžeme začať s animáciou

* 1. Tvorba animácií

Keď máme postavu pripravenú na animáciu – teda správne vymodelovanú, otextúrovanú, s dobrým rigom a upraveným váhovaním – môžeme sa konečne pustiť do samotnej animácie. Blender ponúka rôzne nástroje, ktoré nám umožnia vytvárať plynulé pohyby postáv, predmetov alebo kamier. Základom je **keyframe animácia**, ale môžeme použiť aj **Inverse Kinematics (IK), animáciu pomocou kriviek alebo dokonca motion capture dáta**.

* 1. Keyframe animácia

Blender funguje na **keyframe princípe**, čo znamená, že v určitých časových bodoch nastavíme dôležité polohy objektu alebo časti tela, a Blender automaticky doplní prechody medzi nimi. Keyframy pridávame stlačením **I** a vybratím vlastnosti, ktorú chceme animovať (napr. polohu, rotáciu, mierku).

Príklad: Ak chceme, aby postava zdvihla ruku:

1. Na **frame 1** nastavíme ruku dole a pridáme keyframe.
2. Na **frame 20** zdvihneme ruku hore a pridáme ďalší keyframe.
3. Blender automaticky vytvorí pohyb medzi týmito bodmi.

Všetky keyframy môžeme upravovať v **Timeline (časovej osi)** a presne dolaďovať pomocou **Dope Sheetu** alebo **Graph Editora**, kde vidíme animáciu vo forme kriviek.

* 1. Inverse Kinematics (IK)

Ak by sme animovali ruku alebo nohu manuálne po každej kosti, bolo by to zdĺhavé a nepraktické. Preto používame **Inverse Kinematics (IK)**, kde nastavíme kontrolné body (napr. ruku) a Blender automaticky prispôsobí polohu ostatných častí (napr. lakťa a ramena).

Toto je super užitočné najmä pri nohách – keď nastavíme chodidlo na zem, Blender zabezpečí, že sa pri pohybe tela správne ohnú aj kolená. IK môžeme aktivovať v **Bone Constraints** paneli a pridať **IK solver**, ktorý umožní ovládať končatiny efektívnejšie.

* 1. Looping animácií

Ak robíme herné animácie, často potrebujeme animácie, ktoré sa opakujú – napríklad chôdza, beh alebo dýchanie postavy. Aby sme vytvorili plynulú **looping animáciu**, musíme zabezpečiť, že prvý a posledný keyframe majú rovnakú hodnotu.

Blender nám umožňuje **kopírovať a vkladať pózy** alebo použiť **Graph Editor**, kde môžeme nastaviť tzv. **Cyclic Extrapolation**, vďaka čomu sa animácia plynulo opakuje.

* 1. Facial animácia (mimika)

Ak chceme, aby postava vyjadrovala emócie, musíme animovať jej tvár. Existujú dva hlavné spôsoby:

1. **Shape Keys (Blend Shapes)** – vopred si nastavíme rôzne výrazy (úsmev, zatvorené oči, otvorené ústa) a plynulo ich miešame v animácii.
2. **Rigovaná tvár** – vytvoríme kosti pre ústa, obočie a oči, ktoré ovládajú deformáciu tváre. Toto sa často používa pri animovaných filmoch alebo hrách, kde potrebujeme veľkú kontrolu nad pohybmi.

Pre lepší výsledok môžeme použiť aj **motion capture** alebo napojiť animáciu na audio súbor, aby sa ústa hýbali podľa zvuku.

1. Programovanie v Ue5

Pri vývoji našej hry sme sa sústredili na niekoľko hlavných oblastí: tvorbu terénu, prácu s objektmi a ich materiálmi, ako aj programovanie herných mechaník. Všetko sme realizovali v **UE5**, ktorý nám poskytol široké možnosti a jednoduché ovládanie.

* 1. Tvorba terénu

Najskôr sme vytvorili základné prostredie pomocou **terénneho editora** (nástroj na úpravu povrchu), kde sme si vymodelovali kopce, údolia a rôzne terénne nerovnosti. Použili sme rôzne štetce na úpravu výšky a tvaru krajiny, aby sme dosiahli realistický vzhľad. Následne sme na terén aplikovali **výškovú mapu** (textúra určujúca výšku povrchu), ktorá nám pomohla s detailnejšími nerovnosťami.

Aby herné prostredie nepôsobilo prázdne, pridali sme **vegetáciu** (rastliny a stromy) cez **Foliage systém** (nástroj na umiestňovanie objektov), kde sme mohli rýchlo rozmiestniť stromy, kríky a trávu po celom teréne.

* 1. Objekty a Materiály

Na tvorbu rôznych objektov, ako sú budovy, skaly alebo predmety v hre, sme použili **Blender**. Po vymodelovaní sme ich importovali do Unrealu a nastavili im materiály.

Materiály sme tvorili pomocou **editoru materiálov** (nástroj na úpravu povrchu objektov), kde sme nastavili farbu, odraz svetla a rôzne efekty. Použili sme **textúry** (obrázky povrchu objektov) a pridali rôzne **mapy** (dáta ovplyvňujúce vzhľad), ako napríklad:

* **Normálovú mapu** (mapa nerovností na povrchu) – pridala detailnejší vzhľad bez zbytočného zaťažovania výkonu.
* **Roughness mapu** (mapa drsnosti povrchu) – ovplyvnila, či povrch vyzerá hladko alebo drsne.
* **Metallic mapu** (mapa kovového efektu) – určovala, ktoré časti objektu budú pôsobiť ako kov.

1. Herné mechaniky

Pri tvorbe našej hry sme museli premyslieť a implementovať rôzne **herné mechaniky**, ktoré určujú, ako sa hráč pohybuje, interaguje s prostredím a postupuje hrou. Každá mechanika je dôležitá pre plynulý a zábavný herný zážitok, preto sme ich starostlivo navrhli a doladili tak, aby spolu dobre fungovali.

* 1. Herné mechaniky hráča

Jednou z najzákladnejších mechaník je **pohybová mechanika**. Využili sme zabudovaný systém pohybu v **hernom engine (herný nástroj)** a upravili sme ho podľa potrieb našej hry. Hráč môže chodiť, behať a skákať, pričom sme nastavili rýchlosť a fyzikálne správanie tak, aby pohyb pôsobil prirodzene.

Aby sa hráč mohol zapájať do sveta hry, pridali sme systém **interakcie (spolupráce s objektmi)**. Vďaka tomu môže hráč zbierať predmety alebo aktivovať tlačidlá. Túto funkciu sme naprogramovali pomocou **skriptov (súbor príkazov)**, ktoré kontrolujú, či je hráč v dostatočnej blízkosti a či stlačil správne tlačidlo.

1. Používateľské Rozhranie (UI)

Pri tvorbe našej hry sme venovali veľkú pozornosť **používateľskému rozhraniu (UI – User Interface)**, pretože práve ono hráčovi sprostredkúva dôležité informácie a umožňuje mu jednoduchú interakciu s hrou. Naším cieľom bolo vytvoriť prehľadné a intuitívne **rozhranie (systém ovládania hry)**, ktoré nebude hráča zbytočne rušiť, ale naopak – pomôže mu lepšie sa orientovať.

* 1. Ui

Po spustení hry sa hráč dostane do **hlavného menu (úvodná obrazovka)**, kde si môže vybrať medzi **novou hrou (začiatok od začiatku)**, **pokračovaním (načítanie uloženého postupu)** alebo **nastaveniami (možnosť úpravy zvuku)**. Celé menu sme vytvorili pomocou **grafických prvkov (vizuálne súčasti rozhrania)**, v našom prípade toto rozhranie je reprezentované počtom vykopaného materiálu.

* 1. Zvuky

1. Marketingová stratégia
   1. marketingový plán
2. Zoznam použitej literatúry

[1] Blender foundation.: Blender 4.3 Manual. Amsterdam (2002). [cit. 2025-1-15]. (online). Dostupné na:

<https://docs.blender.org/manual/en/latest/>

[2] Image-Line.: FL Studio Reference Manual. Belgicko (1998). [cit. 2025-1-16]. (online) Dostupné na:

<https://www.image-line.com/fl-studio-learning/fl-studio-online-manual/>

[3] Adobe.: General Knowlage. Kalifornia (2019). [cit. 2025-1-13](online) Dostupné na:

<https://substance3d.adobe.com/documentation/spdoc/substance-3d-painter-174529811.html>

[4] Epic Games.: Unreal Engine 5.5 Documentation. Severná Karolina (1998). [cit. 2025-1-3] (online) Dostupné na:

<https://docs.unrealengine.com/>

[5] GitHub, Inc.: Reference Manual. San Francisco (2007). [2024-11-10]. (online) Dostupné na:

<https://git-scm.com/doc>