



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

POČÍTAČOVÁ HRA PRO BOJ SE SUCHEM V UNITY

A COMPUTER GAME TO FIGHT DROUGHT IN THE COUNTRYSIDE IN UNITY

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

DANIEL OLEARČIN

VEDOUcí PRÁCE

SUPERVISOR

Ing. VÍTĚZSLAV BERAN, Ph.D.

BRNO 2022

Zadání bakalářské práce



Student: **Olearčín Daniel**
Program: Informační technologie
Název: **Počítačová hra pro boj se suchem v Unity**
A Computer Game to Fight Drought in the Countryside in Unity
Kategorie: Uživatelská rozhraní

Zadání:

1. Seznamte se principy vývoje her, podobnými edukativními počítačovými hrami, základními opatřeními proti vysychání krajiny a vývojem aplikací na platformě Unity.
2. Navrhněte simulační hru, která umožní hráči aplikovat v malé lokalitě typu vesnice různá opatření proti vysychání krajiny a na základě jednoduchých modelů a pravidel sledovat dopad těchto opatření. Zaměřte se především na efektivní ovládání a dobrou hratelnost.
3. Navrženou hru implementujte na platformě Unity s využitím vhodných existujících knihoven a doplňků.
4. Řešení průběžně testujte a vylepšujte na reálných uživatelích.
5. Presentujte klíčové vlastnosti řešení formou plakátu a krátkého videa.

Literatura:

- K. Becker, J.R. Parker. The Guide to Computer Simulations and Games 1st Edition. Wiley; 1st edition. ISBN: 978-1118009239.
- Nicholas Lowell. The pyramid of game design: designing, producing and launching service games. CRC Press, 2019. ISBN: 978-1-138-29889-7.
- Tomáš Holan. Unity: První seznámení s tvorbou počítačových her. Edice CZ.NIC.
- Dále dle pokynů vedoucího.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1, 2 a částečně body 3 a 4.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Beran Vítězslav, Ing., Ph.D.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2021

Datum odevzdání: 11. května 2022

Datum schválení: 1. listopadu 2021

Abstrakt

Táto bakalárska práca sa zaoberá návrhom a vývojom hry v hernom engine Unity. Hra má za úlohu byť edukatívna. Hráč má možnosť oboznámiť s rôznymi okolnosťami, ktoré vyplývajú na výnosnosť dediny a zádržnosť vody v krajine. Cieľom hry je mapovanie malej oblasti dediny, nad ktorou je možné vykonávať zamieňanie jednotlivých políček za iné typy. Pomocou jednoduchých modelov, charakteristických výpočtov a pravidiel je možné sledovať dopad týchto zmien na výnosnosť dediny a zádržnosť vody v krajine.

Abstract

This bachelor's thesis focuses on the design and development of the game in the Unity gaming engine. Game has the task of being educational. The player has the opportunity to familiarize himself with the different circumstances, which result on the profitability of the village and the retention of water in the country. The aim of the game is to map a small village above which it is possible to convert individual fields for other types. With simple models, characteristic calculations and rules, it is possible to monitor the impact of these changes on the profitability of the village and the retention of water in the country.

Kľúčové slová

Unity, Unity3D, Blender, C#, herné mechaniky, vysychanie krajiny, vývoj počítačovej hry, vývoj dediny v čase, herný dizajn, edukatívne hry

Keywords

Unity, Unity3D, Blender, C#, game mechanics, drought of the country, computer game development, village development in time, game design, educational games

Citácia

OLEARČIN, Daniel. *Počítačová hra pro boj se suchem v Unity*. Brno, 2022. Bakalárska práca. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Vítězslav Beran, Ph.D.

Počítačová hra pro boj se suchem v Unity

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Ing. Vítězslava Berana. Uviedol som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpal.

.....
Daniel Olearčín
8. mája 2022

Podakovanie

Týmto by som chcel poďakovať pánovi Ing. Vítězslavu Beranovi, za užitočné pripomienky, ochotu, nájdenie si času a usmernenie pri písaní bakalárskej práce.

Obsah

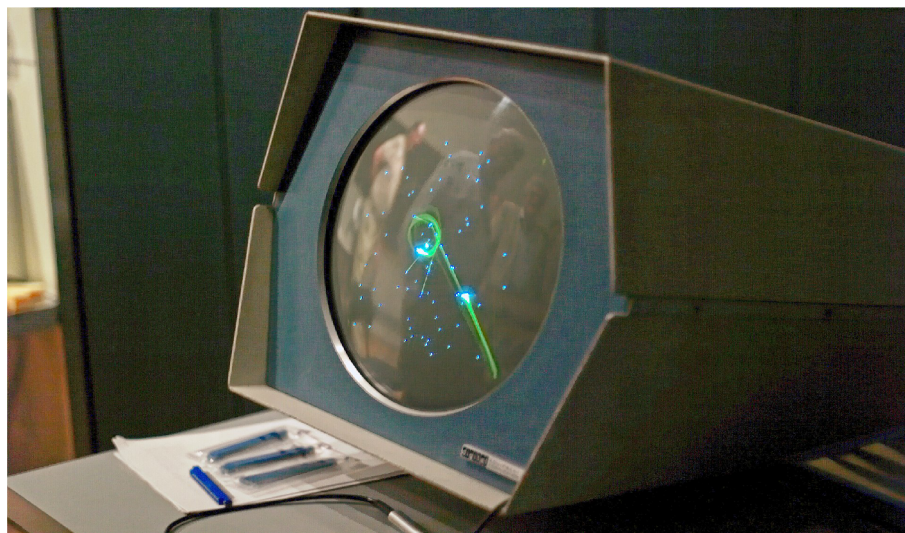
1	Úvod	2
2	Počítačové hry a vysychanie krajiny	4
2.1	Počítačová hra všeobecne	4
2.2	Žánre počítačových hier	5
2.3	Edukatívne hry	6
2.4	Vysychanie krajiny	7
3	Vytváranie počítačových hier	9
3.1	Vytváranie hier	9
3.2	Techniky herného designu	10
3.3	Komunikácia medzi hráčom a hrou	11
3.4	Herný engine unity	12
3.5	Unity visual editor	13
3.6	Porovnanie 2 popredných herných nástrojov na vytváranie hier	15
4	Návrh hry	18
4.1	Požiadavky na hru	18
4.2	Architektúra a užívateľské rozhranie hry	18
4.3	Herné časti a mechaniky	21
5	Implementácia herného menu	24
5.1	Základ herného menu	24
5.2	Skript na ovládanie herného menu	26
6	Implementácia mapy dediny	28
6.1	Zloženie scény	28
6.2	MapController skript	29
6.3	MapMouseMovement skript	34
7	Testovanie	35
8	Záver	37
	Literatúra	38
A	Obsah priloženého pamäťového média	40

Kapitola 1

Úvod

Hranie hier a celkovo hry sa v dnešnej dobe, hlavne medzi deťmi, stávajú nevýhnutnou súčasťou našich životov. Každým dňom na svete vychádza obrovské množstvo hier, niektoré hry sa vyvíjajú pár rokov a niektoré zase pár dní. V minulosti hru väčšinou vyvíjal jeden človek alebo malý tím, v dnešnej dobe to je takmer nemožné. Na väčšine nových populárnych hier, ako sú Valorant a New World, pracuje veľké množstvo tímov aj po vydaní hry do sveta. Nové hry doslova vyžadujú neustále vylepšovanie na základe recenzií hráčov tejto hry, inak by mohli veľmi ľahko zaniknúť z dôvodu chýb, ktoré neboli odhalené pri testovaní hry alebo málo obsahu v hre.

Prvá klasická počítačová hra bola vyvinutá v roku 1962 a volala sa Spacewar!. Bola vyvinutá tímom programátorov pod vedením Steva Russella na Massachusettskom technologickom inštitúte. Jej cieľom bolo zábavnou formou demonštrovať výkonnosť počítača PDP-1. Odvtedy sa vývoj hier drasticky zmenil a dnes už poznáme rôzne platformy na vývoj hier. Medzi najznámejšie patria **Unreal Engine** a **Unity**, v ktorých hru ako bola Spacewar! dokáže programátor, ktorý má množstvo skúsenosti, naprogramovať sám behom jedného dňa.



Obr. 1.1: Obrázok hry Spacewar! na počítači PDP-1 [15].

K počítačovým hrám som mal celý život blízko a veľmi ma zaujíma nielen ich hranie, ale taktiež ich vytváranie. Keď som bol malý, tak bolo mojím snom si jedného dňa vytvoriť počítačovú hru, ktorú budú ľudia hrať po celom svete. Keďže moje štúdium značne zahŕňa programovanie, tak aj ja som si vyskúšal vytvoriť pár jednoduchých hier podľa návodov na internete, ktoré som si doplnil o rôzne zaujímavé doplnky. Napríklad hru piškvorky, ktorú určite všetci poznajú ako typickú strategickú hru pre dvoch, som pomocou hernej platformy Unity pretvoril na hru kde mohli hrať až štyria hráči vzájomne. Keďže aj takéto malé pozmenenie hry ma veľmi bavilo naprogramovať, pri výbere bakalárskej práce som neváhal a vybral som si túto prácu.

Cielom tejto práce je vytvoriť editor mapy dediny, ktorý bude realisticky zobrazovať výnosnosť v dedine a zádržnosť vody v krajine. Zámerom tejto hry je editovanie mapy a súčasné učenie sa a zisťovanie, ktoré zmeny prospievajú dedine na základe rôznych okolností.

V kapitole 2 sa nachádza klasifikácia počítačových hier, zaradenie tejto práce do správneho žánru, opis edukatívnych hier a vysvetlenie ako predchádzať vysychaniu krajiny. Ako správne vyvíjať hry je opísané v kapitole 3, kde je taktiež popísaný herný engine Unity so zameraním na unity visual editor a jeho porovnanie s Unreal engine. V kapitole 4 je opísaný návrh hry a cieľ hry. V kapitolách 5, 6 sa venujem implementácii herného menu a mapy dediny, ktoré sú vyvíjané v dvoch rôznych scénach. Nakoniec v kapitole 7 zhrniem výsledky z môjho testovania.

Kapitola 2

Počítačové hry a vysychanie krajiny

V tejto kapitole bude popísané, čo je to počítačová hra, ako sa hry rozdeľujú v dnešnej dobe a žánery vytvorenej hry. Následne budú opísané edukatívne hry a vysychanie krajiny.

2.1 Počítačová hra všeobecne

„Počítačová hra je interaktívny softvér komunikujúci s používateľom pomocou 2D alebo 3D grafiky a zvukov na účely zábavy alebo vzdelávania [14].“

Každý človek chápe pojem počítačová hra trochu inak. Staršia generácia si prevažne myslí, že je to strata času. Pre mladú generáciu to môže byť spôsob zoznamovania sa s ľuďmi, zabávania sa a dokonca spôsob učenia nových vecí. Napríklad v hre Farming Simulator, sa môže každý naučiť ako sa starať o polia, sadenie a zber rôznych obilnín a nespočetne veľa iných vecí. V dnešnej dobe sa pri hraní hier veľmi často vyžaduje komunikácia s ďalšími hráčmi. Pre dorozumievanie sa používa veľmi rozšírený anglický jazyk, ktorý sa učí aj v školách. Pri komunikácii má hráč možnosť zlepšovať svoju úroveň angličtiny. Hra VR chat ponúka možnosť spoznávať nových ľudí pomocou virtuálnej reality, kde hráči môžu hrať spoločne rôzne hry ako sú napríklad volejbal a futbal.

Všetko má ale svoje nevýhody. Hranie hier môže prerásť do závislosti, kyberšikanovanie v dnešnej dobe je veľmi časté, väčšina hier ponúka obsah ktorý sa dá kúpiť za reálne peniaze a veľa ďalších nevýhod, na ktoré si treba dávať pozor hlavne pri malých deťoch.

Počítačové hry môžu byť taktiež náročné a nie je ich možné hrať na výkonnostne slabších počítačoch. Preto pri každej hre sú uvedené minimálne a odporúčané požiadavky na zariadenie. Medzi najdôležitejšie údaje patrí frekvencia procesora, výkonnosť grafickej karty, potrebné voľné miesto na pevnom disku a kapacita operačnej pamäte. Ak počítač nespĺňa minimálne požiadavky hry, tak s najväčšou pravdepodobnosťou bude hra sekať alebo v horšom prípade sa ani nezapne. Nové hry majú väčšinou vysoké minimálne požiadavky, pretože vzhľad a funkcie hier sa každým rokom posúvajú vyššie.

2.2 Žánre počítačových hier

Žáner udáva stručne charakteristiku hry a čo bude úlohou hráča v danej počítačovej hre. Pomocou neho sa dajú filtrovať hry. Napríklad ak hráč chce hrať športovú hru, tak bude hľadať hry pod žánrom šport. Počítačové hry sa delia na veľa rôznych žánrov. Niektoré hry majú dokonca viacero žánrov ako napríklad hra Warcraft3, ktorá patrí do žánru stratégie a taktiež RPG.

Žáner RPG

RPG je skratkou z anglického výrazu role-playing game. Po slovensky to znamená hra na hranie rolí. Podstatou tejto hry je hranie za nejakú postavu, ktorá má svoju podstatu v dejovej línii. Tieto hry sú obvykle dlhé a veľmi prepracované, majú hlavnú dejovú líniu a veľa vedľajších. Postáv z ktorých si môžeme vybrať je väčšinou viac. Medzi takéto hry patria veľmi obľúbené hry Diablo a Witcher.

MMORPG je skratkou z anglického výrazu massive multiplayer online role-playing games. Sú to hry žánru RPG, ktoré vyžadujú prístup na internet. Hráč sa stretáva s veľkým množstvom hráčov v jednej hre a spoločne sa snažíte túto hru prejsť. V týchto hrách je veľmi dôležitá komunikácia. Výhodou týchto hier je zdokonaľovanie komunikácie a spoznávanie nových ľudí. Príkladom takejto hry je New World.

Žáner Arkády

Arkády boli jedny z prvých hier, ktoré sa dali hrať na monitoroch. Medzi najznámejšie arkády patrí Super Mario a Sonic. Hra sa odohráva v 2D prostredí, kde hráč má za úlohu prejsť level, kde ho čakajú rôzne pasce. Na konci za prejdené kolo väčšinou dostane odmenu v podobe peňazí alebo materiálov, za ktoré si môže kúpiť vylepšenia a pomocou nich prejsť ďalší level. Každý nový level nadobúda väčšiu náročnosť.

Žáner Akčné

U hier ktorý majú žáner akčná, sa stále bojuje pomocou rôznych zbraní alebo hrubou silou. Podstatou týchto hier je samotné bojovanie, porážanie stále silnejších nepriateľov a nejaká malá dejová línia k tomu. V dnešnej dobe sa u akčných hier udáva sub-žáner, keďže ich je veľmi veľa. Medzi takéto sub-žánre patria bojové hry, strieľačky, akčne-dobrodružné hry a množstvo ďalších. Veľmi dobrým príkladom je hra Half-Life, kde prevláda boj a dejová línia slúži len na prepojenie jednotlivých nepriateľov a lokácií.

Žáner Prehliadačové hry

Tieto hry sa hrajú v internetových prehliadačoch. Sú jednoduchšie a využívajú **flash player** a **javascript** na funkcionality a hranie hry. Tieto hry sú bezplatné a majú veľmi malé minimálne požiadavky na počítač. Príkladom takejto hry je online strategická hra Divoké Kmene.

Žáner Detské

Hry, ktoré majú žáner detské, sú určené pre deti väčšinou predškolského veku. Sú kreslené, kde vystupujú rôzne zvieratká alebo postavičky z rozprávok. Tieto hry sú veľmi jednoduché, slúžia na zabavenie dieťaťa alebo naučenie sa mien rôznych zvierat a farieb. Medzi takéto

hry patrí hra Pou, v ktorej ide o kŕmenie a staranie sa o postavičku aby bola šťastná, najedená a mala dostatok pohybu.

Žáner Športové

Sú to hry, ktoré simulujú športové udalosti ako sú napríklad futbalový alebo hokejový zápas. Medzi jednu z najpopulárnejších hier tohto žánru, patrí Fifa. Vývojoví tím hry Fifa, každý rok vydá aktualizovanú verziu hry. Tieto aktualizácie majú za úlohu udržiavať aktuálne zostavy tímov a kondíciu ich hráčov.

Žáner Stratégie

Podstatou týchto hier je budovanie základne, stavba jednotiek, ich vylepšovanie a boj proti nepriateľskej strane. Úlohou hráča je splnenie úloh zadaných na začiatku každej misie. Najčastejšie ide o zničenie nepriateľských opevnení a jednotiek, zabratie určitého územia alebo ťažba určitého množstva surovín. Veľmi známou strategickou hrou je hra Plants vs Zombies.

Žáner Adventúry

U hier ktorý majú žáner adventúra, môžeme očakávať veľmi zaujímavý príbeh. Podstatou týchto hier je prechádzanie herného príbehu, získavanie vecí a rozprávanie sa s ďalšími fiktívnymi postavami. Medzi takéto hry patrí napríklad hra Ghost of Tsushima.

Žáner vytváranej hry

Pri hraní hry je pre hráča potrebné aby rozmýšľal a vychádzal z výnosnosti dediny, ktorá sa mení pri zmene políček na mape dediny. Na základe tejto zmeny hráč zisťuje rôzne okolnosti, ktoré vplyvajú na výnosnosť dediny. Hra je edukatívna a hráč má za úlohu zistiť všetky okolnosti a ich efektívnosť pri výpočtoch jednotlivých výnosností. Na základe tejto charakteristiky som určil žáner hry na stratégiu s prvkami simulácie.

2.3 Edukatívne hry

Edukatívne hry sú vytvárané s cieľom rozvíjať hráčove vedomosti v konkrétnom odvetví [6]. Ide teda o rozvíjanie vedomostí zábavným spôsobom, ktorý je pre väčšinu ľudí príjemnejší a jednoduchší. Postupom času sa edukatívne hry dostávajú do popredia v školách, kde majú za úlohu pomôcť pedagógom s výukou detí a taktiež dospelých ľudí.

Kategorizácia výukových hier

Táto kategorizácia bola vytvorená na základe množstva programov od rôznorodných dodávateľov [6].

- **Miera interaktivity** - Interaktivita je veľmi dôležitá z hľadiska výučby. Jedná sa o vzájomnú komunikáciu, pri ktorej sa študent aktívne zapája. Čím viac interaktívnych prvkov, tým lepšie.
- **Úroveň vzdelávania** - Rozlišujeme druhy vzdelania na materské školy, základné školy, stredné školy a vysoké školy.

- **Počet užívateľov** - Pre jedného užívateľa a pre viac užívateľov. Jeden výukový program môže naraz využívať viaceru užívateľov, kde užívatelia si navzájom pomáhajú a radia sa.
- **Počet didaktických funkcií** - Každá výuková hra musí obsahovať aspoň jednu didaktickú funkciu.
- **Tematický rozsah** - Výukové programy môžu obsahovať viaceru tematických celkov alebo iba jeden. Z tohto hľadiska sa delia výukové hry na monotematické a polytematické.
- **Možnosť vnímania** - Výuková hra môže používať zvukové prvky a grafické prvky. Z tohto hľadiska sa delia výukové hry na audiovizuálne a vizuálne.
- **Jazykové mutácie** - Počet jazykov, ktorý sa využíva pri výukovej hre.
- **Verzia** - Demo verzia na vyskúšanie alebo plná verzia, ktorú je potrebné si predplatiť.

Edukatívna počítačová hra Stellarium

Stellarium je počítačové planetárium, ktoré zachytáva oblohu viditeľnú voľným okom a ďalekohľadom [5]. Umožňuje sledovať rôzne astronomické úkazy, planéty, hviezdy a množstvo ďalšieho vo vesmíre. Pre vysvetlenie reálnych úkazov sa používajú skripty, ktoré sa dajú samostatne vytvoriť alebo stiahnuť na internete. Stellarium poskytuje zdrojový kód pre verejnosť, to znamená že každý sa môže podieľať na vývoji a zlepšení tohto programu.

Zdokonaľovanie schopnosti žiaka orientovať sa na nočnej oblohe sa pomocou tejto edukatívnej hry stáva zábavné. Študent si môže zobrazovať jednotlivé súhvezdia a zistiť o nich informácie. Taktiež sa tu nachádza možnosť zadať miesto z kade sa obloha pozoruje.

2.4 Vysychanie krajiny

Vysychanie krajiny je veľmi nebezpečné a má potenciál ohroziť životy ľudí, škodu na majetku a poškodiť ekosystém [7].

Medzi Piliere ako zachrániť krajinu pred vysychaním patrí:

Vytvorenie informačnej platformy o vysychaní

Prvým krokom pri zvládaní rizík je vytvorenie informačnej platformy. Táto platforma monitoruje stav vodných zdrojov a riziká vysychania. Na monitorovanie stavu sú potrebné monitorovacie siete hydrometeorologického ústavu. Veľkým krokom dopredu by bolo vytvorenie informačného systému, ktorý by predpovedal na základe dát vysychanie dediny a predom by na to varoval.

Rozvoj vodných tokov

Druhým krokom je pozorovanie vodných tokov a ich prípadný rozvoj. Predstavuje reakciu na nepriaznivé pozorovania ako sú napríklad zníženie hladiny vody a čistota dostupných vodných zdrojov. Reakcia sa môže odohrávať pomocou realizácie nových vodných zdrojov. Nové vodné zdroje môžu byť napríklad viacúčelové nádrže a umelé infiltrácie povrchových vôd do vôd podzemných.

Polnohospodárstvo

Medzi požiadavkami na vodu v poľnohospodárstve a lesníctve a dostupnými vodnými zdrojmi v krajine by mala vládnuť rovnováha. Typickými príznakmi zhoršenia rovnováhy sú zhoršené stavy lesných porastov, zhoršená kvalita vypestovaných obilnín a znečistenie vody látkami, ktoré prospievajú výžive obilnín. Tieto príznaky sa riešia v rámci tohto piliera pomocou zvýšenia organických látok v pôde a opatrenia na zvýšenie čistoty vodných zdrojov.

Obnova prírodných vodných režimov

Tento pilier monitoruje a následne napravné všetky nepriaznivé dôsledky systematickej straty vody vo vodných tokoch a nepriaznivému zásahu človeka do vodných tokov. Opatrenia v rámci tohto piliera sa hlavne sústreďujú na zvýšenie zadržovania vody v krajine, obnova prirodzených funkcií vodných tokov, obnova porastov a lesných pôd.

Zodpovedné zaobchádzanie s vodou

Tento pilier je zameraný na správne využívanie a zaobchádzanie s vodou. Možností recyklácie a následného čistenia je množstvo, no málokto si dá tú námahu to dodržiavať. K dispozícii je množstvo nových technológií a výskumov ako recyklovať vodu. Tieto technológie môžu výrazne prospievať proti rizikám vysychaniu krajiny.

Technológie pre recykláciu vody sa významne líšia podľa znečistenia vody. Iné požiadavky sa kladú na vodu určenú pre pitie a na vodu určenú pre zavlažovanie plodín [11]. Častým recyklovaním v dnešnej dobe sa stáva zber dažďovej vody a jej následná filtrácia.

Kapitola 3

Vytváranie počítačových hier

V tejto kapitole bude opísaný proces vytvárania hier, techniky herného designu a komunikácia medzi hráčom a hrou. Následne bude popísaný herný engine Unity a jeho porovnanie s konkurenčným herným enginom.

3.1 Vytváranie hier

Na začiatku musí byť dobrý nápad, niečo čo robí hru inovatívnou a bude priťahovať široké publikum hráčov. Môže to znamenať vytvorenie úplne nového herného konceptu alebo len starý, dobre zavedený koncept s niekoľkými inovatívnymi vylepšeniami a prvkami. Tvorba hier sa skladá z plánovania hry a jej samotného vytvárania [3].

Plánovanie hry

Plánovanie je základom úspechu vytvorenia výborného produktu [2]. Výhodami plánovania sú pripravenosť, usporiadanosť, produktivnosť, systematickosť a menší stres.

Plánovanie hry sa skladá z:

- **Zváženie vlastných zručností** - Vytváranie počítačových hier vyžaduje mnoho zručností. Medzi tieto zručnosti patrí programovanie, návrh, práca so zvukovými stopami a efektami, modelovanie, písanie príbehov a množstvo ďalších. Najlepším spôsobom ako si vybrať typ hry je zistiť zručnosti a podľa toho navrhnuť hru.
- **Vytvorenie návrhu hry** - Pred začatím vytvárania hry je veľmi dôležité samotnú hru navrhnuť. Tento návrh by mal obsahovať všetky podrobnosti o hre, základné herné mechaniky, koncepčné umenie, návrh úrovní a štýl tvorby skriptov. Pri práci v tíme sa tento krok stáva mnohonásobne dôležitejší, kvôli správne pochopeniu hry medzi celým tímom.
- **Vybratie herného enginu** - Herný engine je základom tvorby hry. Je to aplikácia, ktorá sa stará o grafické vykresľovanie, spracovanie zvuku, zachytávanie užívateľského vstupu, fyzické vlastnosti objektov pomocou skriptov a množstvo ďalších činností. Existuje viacero herných enginov z ktorých si je možno vybrať. Medzi populárne herné enginy patrí Unity, Unreal Engine, Amazon Lumberyard, Game Maker Studio a Godot.

Vytváranie hry

Proces vytvárania produktu často znamená množstvo ťažkých, smutných ale aj radostných chvíľ.

Vytváranie hry sa skladá z:

- **Plánovanie tvorby projektu** - Tento krok zahŕňa vytvorenie zoznamu všetkých úloh, ktoré je potrebné dokončiť. Tieto úlohy sa rozdeľujú podľa priority a náročnosti dokončenia. Ďalej je potrebné vytvoriť harmonogram, podľa ktorého sa budú jednotlivé úlohy dokončovať. Pri práci v tíme je dôležité si naplánovať opakované stretnutia, na ktorých sa povie čo sa dokončilo, kde nastali problémy a na čom sa pracuje.
- **Vytvorenie prototypu** - Vytvorenie prototypu by nemalo trvať príliš dlho. Prototyp by mal zahŕňať základnú funkcionality hry bez rôznych graficky náročných zobrazovaní a špeciálnych funkcií. Vďaka prototypu sa dá zistiť veľa nových znalostí na základe skúšania a hrania. Následne sa môže prototyp upravovať, až sa z neho stane pevný základ na ktorom sa buduje zvyšok hry.
- **Vytvorenie aktív hry** - Aktíva sú všetko, z čoho sa hra skladá. Patria sem rôzne modely, animácie, grafický dizajn, zvukové efekty a stopy, príbeh a ďalšie pre hru typické dáta. Tieto aktíva sa vyrábajú samostatne a pomocou herného enginu sa využívajú už v konkrétnej hre.
- **Testovanie** - Vykonávať testovanie je veľmi dôležité počas celého vývoja hry. Jednotlivé testovania nám dávajú spätnú väzbu, pomocou ktorej môžeme odstrániť vyskytujúce sa chyby a hru upraviť. Veľa informácií o hre sa dá získať aj jej hraním alebo pozeraním na hráčov, ktorý túto hru hrajú.
- **Optimalizovanie** - Optimalizácia hry spočíva v jej zrýchlení a zlepšení užívateľského rozhrania. Hra by mala byť intuitívna a počet snímkov za sekundu by sa nemal výrazne meniť.
- **Uverejnenie a vydanie** - V prípade ak má byť vyvíjaná hra vydaná pre široké publikum a nie je vytváraná len na vlastné účely, je potrebné si hľadať webovú stránku pre digitálnu distribúciu. V dnešnej dobe už nie je veľký problém dostať svoju hru na známu platformu, ako je Steam. Pred uvedením hry je potrebné urobiť dobrú reklamu, tak aby sa o vydaní hre dozvedelo čo najväčšie publikum.

3.2 Techniky herného designu

Techniky herného designu využívajú množstvo spôsobov ako zaujať a udržať hráča pri hraní hry. Pri týchto spôsoboch sa často využívajú informácie z iných odborov ako je napríklad psychológia [4].

Flow

Pojmom **flow** sa opisuje stav, do ktorého sa hráč dostáva pri hraní hry. Do **flowu** sa hráč dostane, ak sa plne sústreďí na hru a na prekonávanie herných prekážok. Cieľom je teda udržať hráča v tomto stave čo najdlhší čas.

V praxi sa hráč dostáva do tohto stavu, ak je hra dobre navrhnutá a obtiažnosť herných prekážok sa neustále stupňuje. Je nutné na začiatok hry vybrať ideálnu obtiažnosť prvej

hernej prekážky, tak aby ju dokázal zvládnuť každý hráč. Takáto herná prekážka sa nazýva tutorial. Tutorial má za úlohu ukázať hráčovi herné mechaniky a cieľ hry.

Ak má byť hráč čo najdlhšie v tomto stave, hra musí mať jasný cieľ a pokroky. Jednotlivé herné prekážky musia byť jasné a zreteľné.

Zábava

Pri vytváraní jednotlivých herných prekážok je dôležité brať ohľad na zábavu pri ich plnení. Ak sú niektoré herné prekážky nudné, môže to mať zlý vplyv na celkovú hru. Príliš zložité výpočty a mechaniky, ktoré nie sú úplne jasné, majú tendenciu hráčovi zhoršiť zážitok z hrania. Zábava ma teda prednosť pred rôznymi zložitými funkciami, ak to nie je kvôli požiadavkám a nutnosťou hry.

Obťažnosť

Obťažnosť hry je veľmi dôležitým témom. Každý hráč má iné skúsenosti a schopnosti, ktoré môžu ovplyvňovať zážitok z hry. Pre hráča, ktorý má veľa skúseností s podobnými žánrami hry môže byť hra veľmi ľahká a teda nezaujímavá. Platí to aj naopak pri hráčovi, ktorý nemá veľa skúseností a hra bude pre neho veľmi ťažká na prekonanie. Úroveň počiatočnej obťažnosti je v množstve hrách nastaviteľná tak aby neskúsený hráč si mohol obťažnosť zmenšiť a skúsený zvýšiť.

Najčastejším typom obťažnosti pri hrách je lineárna obťažnosť. Ide o obťažnosť, ktorá lineárne stúpa spoločne s progresom hráča v hre. Prvé herné prekážky sú veľmi ľahké a majú za úlohu hráča oboznámiť s hernou mechanikou a cieľom hry. Posledné herné prekážky sú najťažšie a hráč pri nich zvyčajne strávi najväčšie množstvo času.

Menej časté je dynamické vyváženie obťažnosti. Hry s týmto typom obťažnosti sa snažia prispôbiť obťažnosť na základe hráčovho postupu v hre. Umelá inteligencia zisťuje rýchlosť postupu a množstvo chýb, ktoré hráč vykonáva. Ak hráč nemá problém prekonávať herné prekážky tak sa obťažnosť automaticky zvyšuje až pokým mu to začne robiť problém. Pri hráčovi, ktorému trvá dlho prejde hernej prekážky sa obťažnosť automaticky zníži tak, aby to bolo preňho ľahšie.

3.3 Komunikácia medzi hráčom a hrou

Komunikácia medzi hráčom a hrou je veľmi dôležitá pri uvádzaní hráča do deja hry. Uvedenie do deja nie je jednoduchý proces, ktorý by sa dal prezentovať iba jedným spôsobom [8]. Dej hry môže mať viacero línií, ktoré je potrebné rozlišovať. V hrách sa nachádza väčšinou jedna hlavná línia a množstvo vedľajších línií. Do základnej komunikácie medzi hráčom a hrou patrí:

Manuál

V manuáli sa nachádzajú všetky podstatné informácie, ktoré by hráč mal vedieť pred začatím hry. Medzi základné informácie v manuáli patrí popis dejovej línie, návod k ovládaniu a popis hry. V súčasnosti sa manuáli strácajú a sú iba v pozadí. Hráč je oboznámený s ovládaním a dejovou líniou počas hrania hry.

Intro

Úvodné informácie a zoznámenie sa s hrou sa zvyčajne vykonáva pomocou *intra*. Hráč sa oboznámi so základnou mechanikou a dejom hry pomocou obrázkov a animácií.

Rozhovory

Pomocou rozhovorov v hre sa hráč dozvedá ako postupuje v debovej línii hry a čo je potrebné k dokončeniu ďalšej hernej prekážky. Môže ísť o rôzne rozhovory s fiktívnymi postavami alebo nájdením útržkov papiera, videozáznamov a denníkov.

Animácie

V hre odohrávajú rôzne animácie, ktoré spestrujú pôžitok z hry a posúvajú ďalej herný postup. Takéto animácie majú za úlohu udržať hráčovú pozornosť.

3.4 Herný engine unity

Herný engine Unity je aplikačný rámec, pomocou ktorého si každý môže zlahčiť proces vytvárania ale aj udržiavania hier. Tento herný nástroj bol vyvinutý spoločnosťou Unity Technologies. Prvá verzia bola predstavená v roku 2005 a bola podporovaná iba na platforme OS X od spoločnosti Apple. V roce 2021 Unity stále dominuje na trhu mobilných hier, viac ako 50% mobilných hier je vytvorených na hernom engine Unity. Unity ale nie je využívaný iba pri vytváraní a vývoji hier. V automobilovom priemysle sa Unity využíva pre vytváranie modelov nových automobilov vo virtuálnej realite, zostavenie virtuálnych montážnych liniek a výcviku nových zamestnancov. Ďalšie odbory na ktorých sa Unity Technologies sústreďujú sú architektúra, strojnictvo a stavebníctvo. Do roku 2016 sa dala licencia na unity zakúpiť jednorazovým poplatkom. Od roku 2016 prešla Unity Technologies na model predplatného na určitý čas. V súčasnosti Unity ponúka viacero možností predplatných plánov, z ktorých jedna je bezplatná pre osobné použitie jednotlivcov alebo malých spoločností, ktoré nevytvárajú ročné príjmy vo výške viac ako 100 000 dolárov [16].

Multiplatformový nástroj

Novodobý unity visual editor Unity 2021.3 LTS, je momentálne podporovaný na rôznych známych operačných systémoch ako sú Microsoft Windows, Linux a macOS. Pomocou unity visual editoru je možnosť vytvárať a vyvíjať aplikácie na viac ako dvadsiatich piatich rôznych platformách. Medzi najznámejšie patria:

- Android
- Linux
- macOS
- Xbox Series X
- PlayStation 5
- iOS

Kľúčové komponenty

Každý herný engine sa skladá z komponent, ktoré majú za úlohu riešiť rôzne herné funkcionality [16].

Medzi kľúčové komponenty herného enginu Unity patria:

- **Vykresľovací engine** - Táto časť sa stará o renderovanie, vykresľovanie reálne vyzerajúcich objektov na základe spracovania dát grafickou kartou. Pokročilejší užívatelia majú možnosť tento proces meniť pomocou skriptovania a dosiahnuť ultra-realistickú HD grafiku.
- **Fyzikálny engine** - Slúži k simulovaniu fyzikálnych zákonov v hrách. Medzi takéto zákony patrí gravitácia, pohyb postáv, pôsobenie síl a rôzne ďalšie. K tomu, aby sa mohlo pracovať s objektom pomocou fyzikálneho enginu, je potrebné priradiť k príslušnému objektu komponentu `rigidbody` v 3D scéne a `rigidbody2D` v 2D scéne.
- **Zvukový engine** - Tento engine má za úlohu prehrávať, dekomprimovať a načítať zvukové stopy. Unity okrem toho ponúka pokročilejšie funkcie, ako je integrácia viacerých rôznych zdrojov zvuku pomocou audiomixeru.
- **Skriptovanie** - Pomocou skriptov sa vytvárajú herné mechaniky. Toto je časť, ktorá popisuje a kontroluje logiku hry. Napríklad skripty môžu byť použité na pridanie, odstránenie a zmenu rôznych objektov v scéne, kontrolu súladu s pravidlami hry, ukladanie dôležitých hodnôt, ovládanie kolízií objektov, čítanie vstupov z prehrávača a podobne. Programovací jazyk pre písanie skriptov v Unity je C#. Je to objektovo orientovaný, statický programovací jazyk vyvinutý spoločnosťou Microsoft. V niektorých verziách Unity z roku 2017 a staršej boli stále podporované Boo a UnityScript (verzia JavaScript for Unity). Podpora týchto jazykov však bola ukončená v roku 2017, takže C# je hlavný skriptovací jazyk v Unity. Unity poskytuje používateľom veľmi rozsiahlu dokumentáciu celého **Scripting API**.

3.5 Unity visual editor

V tejto časti je popísané jednoduché narábanie s Unity editorom a jednotlivé časti užívateľského rozhrania [13].

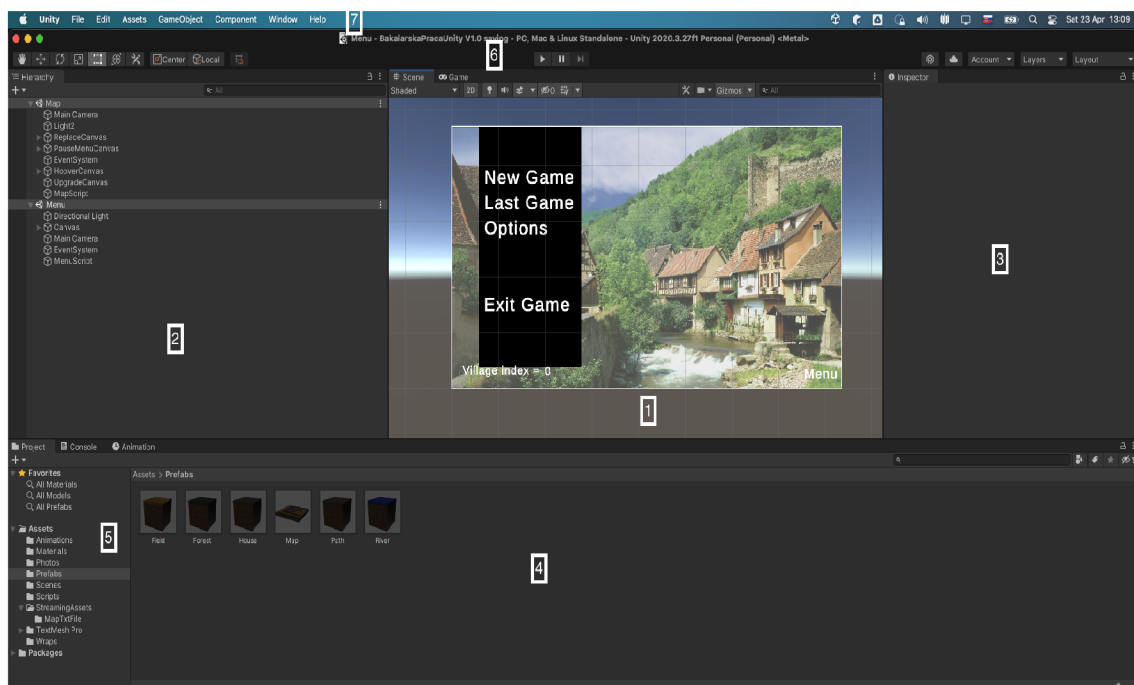
Užívateľské rozhranie unity

V unity je možné vytvárať 2D scény a 3D scény.

Na obrázku 3.1 sa nachádzajú kľúčové okná unity visual editora:

- **1** - Uprostred okna aplikácie sa nachádza **editor scény**. Pomocou neho je možné vytvárať a upravovať scénu. Základné prvky scény sa nazývajú `GameObject`. Tieto prvky je možné po scéne posúvať, rotovať, zväčšovať. Nad editorom scény sa nachádza malá lišta, pomocou ktorej sa dá narábať so zvukom, svetlom, zapínať a vypínať 3D scénu alebo zobrazíť si scénu z pohľadu hernej kamery (hráča hry).
- **2** - Ďalším kľúčovým oknom je **hierarchia objektov**. Nachádza sa na ľavej strane okna aplikácie. V tejto časti sa nachádzajú práve otvorené alebo iba načítané scény, s ktorými sa pracuje. Tieto scény sú reprezentované v textovej podobe a zobrazujú čo sa v scéne nachádza. Jednotlivé scény je možné si otvoriť a zistiť s akých objektov sa skladajú.

- **3** - Na pravej strane okna aplikácie sa nachádza okno **Inšpektor**. Pri zvolení rôzneho objektu pomocou hierarchie objektov alebo editora scény sa otvorí jeho konfigurácia. V konfigurácii objektu sa nastavujú rôzne vlastnosti alebo pridávajú komponenty vzhľadom na to o aký objekt sa jedná. Taktiež v tomto okno sa upravujú **prefabrikáty**. Prefabrikáty sú konkrétne konfigurácie herných objektov, ktoré sú uložené z pravidla v položke **Prefabs**. Vytvárajú sa kvôli opätovnému používaniu v scéne alebo scénach.
- **4** - Na spodnej časti okna aplikácie sa nachádza okno projektu. V ňom je možné vidieť všetky konkrétne súbory, ktoré sa nachádzajú v otvorenej zložke v štruktúre projektu. Následne je možné ich z tadeto otvoriť a upravovať v zvolenom editore (skripty) alebo ich otvoriť v okne inšpektora (prefabrikáty).
- **5** - V ľavom rohu okna aplikácie sa nachádza štruktúra projektu. V tejto štruktúre sa nachádzajú všetky potrebné súbory pre správny beh aplikácie alebo predom vytvorené súbory, ktoré budú v budúcnosti využité v nejakej scéne.
- **6** - Pomocou tejto lišty, ktorá sa nachádza na vrchnej časti, sa riadi beh aplikácie v editore a rôzne možnosti úpravy scény. V strede tejto lišty sa nachádzajú 3 tlačítka, pomocou ktorých je možné spustiť, zastaviť a vypnúť hru. Na pravej strane tejto lišty sa nachádzajú informácie o práve prihlásenom účte a vrstvách scény. Taktiež je tu možnosť meniť vybrané rozloženie editora. Na ľavej strane tejto lišty sa nachádzajú rôzne nástroje pre prácu s objektmi v scéne.
- **7** - Tejto lište, ktorá sa nachádza na absolútnom vrchole stránky sa hovorí hlavná lišta. Pomocou nej je možné načítať alebo uložiť projekt, meniť nastavenia projektu a množstvo ďalších nastavení editora.



Obr. 3.1: Základné užívateľské rozhranie unity visual editora.

Životný cyklus skriptov

Každý skript má svoj vlastný životný cyklus. Tento cyklus môže byť ovplyvnený rôznymi událostnými funkciami, ktoré nám unity ponúka [12].

Medzi základné udalostné funkcie, ktoré patria ku práci s unity enginom, patria:

- **Awake()** - Táto funkcia je volaná ako prvá zo všetkých. Telo funkcie sa vykoná v momente kedy začne byť **GameObject** alebo **Scene** ku ktorej je pripojený skript aktívny.
- **Start()** - Telo funkcie sa vykoná iba raz a to na začiatku spustenia skriptu. Táto funkcia sa používa na nastavenie predvolených hodnôt.
- **Update()** - Táto funkcia sa opakovane volá. Každý snímok v hre znamená jedno prevedenie tela funkcie.
- **LateUpdate()** - Funkcia má rovnakú úlohu ako **Update()**, s rozdielom že sa volá až po prevedení funkcie **Update()**.
- **FixedUpdate()** - Táto funkcia sa využíva, ak je potrebné volať funkciu častejšie ako raz za jeden snímok. Frekvencia volania funkcie nesúvisí s počtom snímkov ale udáva sa v konštantnom čase.
- **OnEnable()** - Funkcia sa zavolá v momente ak sa stane **GameObject** spoločne s pripojeným skriptom aktívny.
- **OnDisable()** - Funkcia sa zavolá v momente ak sa stane **GameObject** spoločne s pripojeným skriptom neaktívna.
- **OnDestroy()** - Ak dôjde k odstráneniu objektu, ktorý môže byť odstránený pomocou funkcie **Object.Destroy()**, prevedie sa telo tejto funkcie.

3.6 Porovnanie 2 popredných herných nástrojov na vytváranie hier

V Nasledujúcej časti sú vybraté v krátkosti opísané herné enginy a vybraté ich kľúčové vlastnosti. Nakoniec je napísané v čom sa tieto herné enginy líšia a situácie, ktoré ukazujú aký herný engine si vybrať na vývoj hry [1][10].

Unreal engine

Unreal Engine je kompletná sada vývojových nástrojov, ktoré sú určené pre každého, kto pracuje s technológiou v reálnom čase. Je to zdrojový herný engine. Pomocou neho je možné vytvárať vysoko kvalitné hry naprieč konzolami, počítačmi, mobilnými telefónmi, virtuálnou realitou a vylepšenou realitou. Zahŕňa to bojové hry a masívne multiplayerové online hry. Kľúčové vlastnosti:

- Je to zdrojovo dostupný herný engine a bol vyvinutý spoločnosťou Epic Games.
- Prvýkrát predstavený v roku 1998, strieľačka prvej osoby Unreal.
- Používa C++ alebo JavaScript. Niektorí ľudia si myslia, že C++ je archaický, zatiaľ čo iní to prisahajú.

- Vysoko kvalitné hry na počítačoch, mobilných telefónoch, notebookoch, konzolách...
- Robustný multiplayerový rámec, VFX a simulácia častíc, flexibilný editor materiálov, rozsiahla sada nástrojov na animáciu...
- Najnovšou stabilnou verziou je momentálne Unreal Engine 5.

Unity

Unity sa využíva v 34% hier, ktoré hráme a prosperujú v komunite. Popredný herný engine na svete, ktorý sa používa na vytváranie polovice svetových hier. Unity je multiplatformový herný engine. Jeho pomoc pri mnohých projektoch znamená, že hry budú rýchlejšie vyvíjané a dosiahne sa vysoká kvalita.

Kľúčové vlastnosti:

- Je to multiplatformový herný engine a bol vyvinutý spoločnosťou Unity Technologies.
- Prvýkrát oznámené a vydané v júni 2005 spoločnosťou Apple Inc.
- Naprogramovaný pomocou C# (Boo) alebo dialektom jazyka JavaScript. Boo je jazyk príkazové riadka podobný jazyku Python.
- Používa sa na vývoj videohier pre webové doplnky, doplnky pre stolné počítače, konzolu a mobilné zariadenia.
- Vylepšenia 2D, animácie, vylepšenia platformy, vylepšenia zvuku a vytváranie snímkov.
- Najnovšou stabilnou verziou je momentálne Unity 2021.3 LTS.

Unreal Engine vs Unity

Pri vývoji hier je veľmi dôležité vybrať správny herný engine [9][17].

Oba tieto herné engine môžu produkovať ohromujúcu úroveň grafiky ale unreal engine sa považuje za lepší v tomto ohľade. Unreal engine má k dispozícii množstvo nástrojov, ktoré fungujú okamžite a nie je potrebné ich vyladovať. Príkladom takýchto nástrojov sú **volumetric lights**, **post-process**, **lens flares**, ktoré sú zabudované v hernom engine a predstavujú vysokú úroveň. Obrovskou výhodou práce s Unreal enginom je jeho svetelný systém, vďaka ktorému vyzerajú všetky objekty veľmi realisticky. Unity engine môže taktiež poskytnúť skvelé výsledky ohľadom grafiky. Unity neustále pracuje na zlepšovaní vizuálnej úrovne motora. S dostatočnými znalosťami sa dajú dosiahnuť podobnú úroveň detailov ako v Unreal engine.

Veľmi dôležitým faktorom pri vyberaní herného engine je taktiež typ hry [17].
Základné typy hier:

- **2D** - V prípade 2D hier je v popredí unity engine. Unity engine sa sústreďí na vývoj 2D hier a má oveľa väčší počet nástrojov na prácu s 2D scénami v porovnaní s Unreal engine.
- **3D** - Obidva herné engine sú schopné vytvárať 3D hry. V prípade jednoduchých 3D hier sa odporúča Unity engine. Pre hry, ktoré majú vysoké nároky na grafiku sa odporúča Unreal engine.
- **Virtuálna reality** - Pre tento typ hier jednoznačne vyhráva Unity engine, keďže je multiplatformový.
- **Hra pre viacerých hráčov** - Obidva herné engine sú schopné vytvárať hry pre viacerých hráčov. Unreal engine je lepšou voľbou pretože má pre tento typ hier integrovanú podporu, zatiaľ čo u Unity engine sa to musí riešiť pomocou dodatočných nástrojov.
- **Mobilná hra** - Unity engine je považovaný za najlepší nástroj na vytváranie mobilných hier, pretože podporuje množstvo vývojových prostredí zároveň.

Ďalším dôležitým faktorom je náročnosť používania herného engine [17]. Učenie sa pracovať s herným engine nie je ľahká záležitosť. Unity engine sa považuje pre začiatočníkov za jasnú voľbu. Má priateľské užívateľské rozhranie, ktoré je intuitívne. Unreal engine má na druhú stranu na začiatok veľmi náročné užívateľské rozhranie. Veľkú váhu má aj písanie skriptov. Unity engine používa na písanie skriptov C# a Unreal engine používa C++. C# má ľahšie pochopiteľnú syntax v porovnaní s C++.

Kapitola 4

Návrh hry

V tejto kapitole je čo najviac objasnený postupný návrh a plán vytvárania hry. Návrh je základom pri vytváraní všetkých rôznych aplikácií a preto je potrebné si pri návrhu dať záležať a nepodceniť to.

4.1 Požiadavky na hru

Hra má byť edukatívna a teda najväčší prínos bude mať ak sa jednotliví hráči niečo naučia. Prostredníctvom tejto hry sa hráč učí a zisťuje informácie ohľadom okolností, ktoré vedú k zvýšeniu výnosnosti dediny. Hráč má možnosť získavať všetky dôležité informácie pre výpočty výnosnosti dediny.

Konkrétne okolnosti a ich dôležitosť hráč nevie. K zisteniu týchto okolností a ich efektívnosti bude musieť hráč meniť jednotlivé políčka a zamýšľať sa ako vplývajú jednotlivé zmeny na výnosnosť a podľa toho zisťovať okolnosti. Podľa týchto okolností sa pomocou výpočtov získava výnosnosť dediny.

Pri hraní hry nemá byť príliš náročné zistiť všetky okolnosti, pretože by sa hráč toho za ten čas veľa nenaučil. Naopak nemôže byť ani príliš jednoduché, pretože by to znamenalo príliš primitívne výpočty a málo existujúcich okolností. Ideálna výnosnosť u dediny sa považuje od 90 a vyššie. Hráčovi, ktorý nie je oboznámený od začiatku so všetkými okolnosťami a ich efektívnosťou, by dosiahnutie takéhoto výsledku nemalo trvať menej ako 4 minúty a viac ako 25 minút pri rôzne vytvorenej dedine. Čas dosiahnutia ideálneho výsledku lineárne stúpa spoločne s veľkosťou vytvorenej mapy.

Hra nemôže byť výkonnostne náročná, dôležité je aby hra bola spustiteľná a hrateľná na rôzne výkonných zariadeniach.

4.2 Architektúra a užívateľské rozhranie hry

Pri návrhu je dôležité si určiť na akej ako sa hra bude ovládať, cieľ hry, ako hra bude zložená a jej užívateľské rozhranie. Všetky tieto časti budú opísané v tejto sekcii.

Architektúra hry

Hra bude vytváraná v hernom engine `Unity` v podobe počítačovej hry. Keďže herný engine `Unity` podporuje rôzne platformy, nie je určené pre ktorú platformu sa hra výsledná aplikácia vyvíja. Možným výberom je operačný systém `Windows`, `macOS` alebo `Linux`. Hra sa

spúšťa pomocou vytvoreného spustiteľného súboru, ktorý je vytvorený pomocou herného enginu Unity.

Požiadavky na počítač na ktorom je hra spúšťaná budú minimálne. Hra nevyžaduje žiadne náročné výpočty alebo grafické animácie, kvôli ktorým by mali byť zdvihnuté požiadavky. Ak to bude aj tak potrebné, bude v počiatočnom menu možné zmeniť úroveň detailov, rezolúciu a režim na celú obrazovku.

Mechanika hry

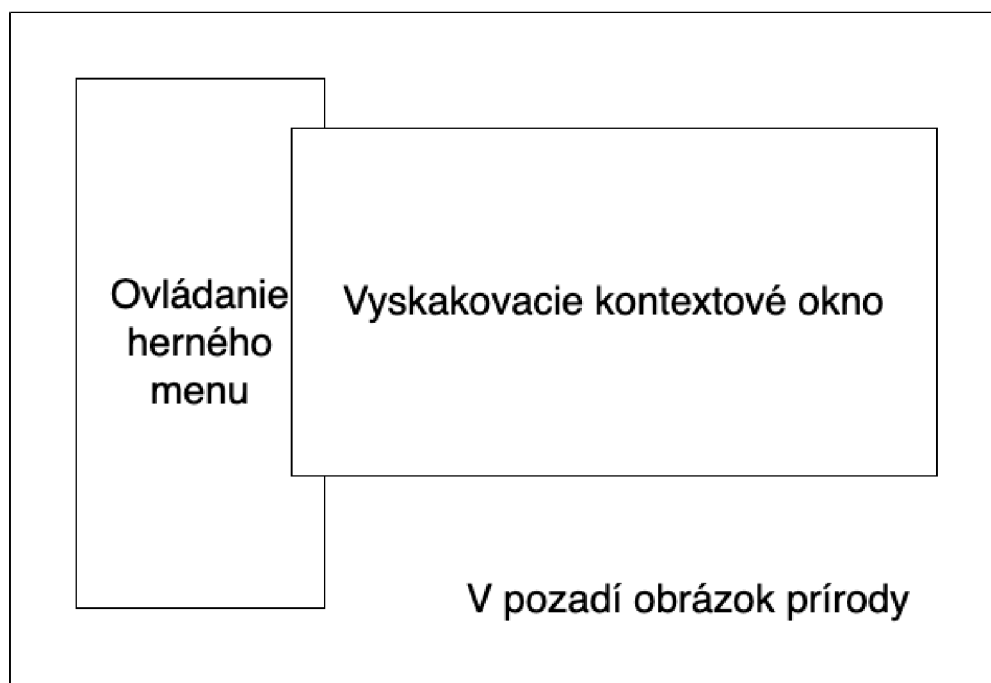
Hlavné ovládanie hry sa vykonáva pomocou počítačovej myši alebo touchpadu. Pomocou ľavého tlačidla na myši alebo touchpadu, bude hráč schopný ovládať celú funkcionality hry klikaním na jednotlivé tlačítka v spustenej hre. Potrebná bude taktiež klávesnica, pomocou ktorej si hráč definuje mapu v pomocou textového súboru alebo v hre, pomocou vstupného textového poľa.

Užívateľské rozhranie hry

Hra by mala pozostávať z dvoch dôležitých scén. Jedna scéna bude určená pre herné menu a druhá pre mapu dediny.

Scéna pre herné menu sa stáva aktívnou hneď po spustení hry. Obsahuje herné menu, pomocou ktorého môže hráč vytvoriť novú mapu, načítať uloženú mapu, vypnúť hru a meniť grafické alebo zvukové nastavenia hry. O funkcionality tejto scény sa bude starať samostatný skript.

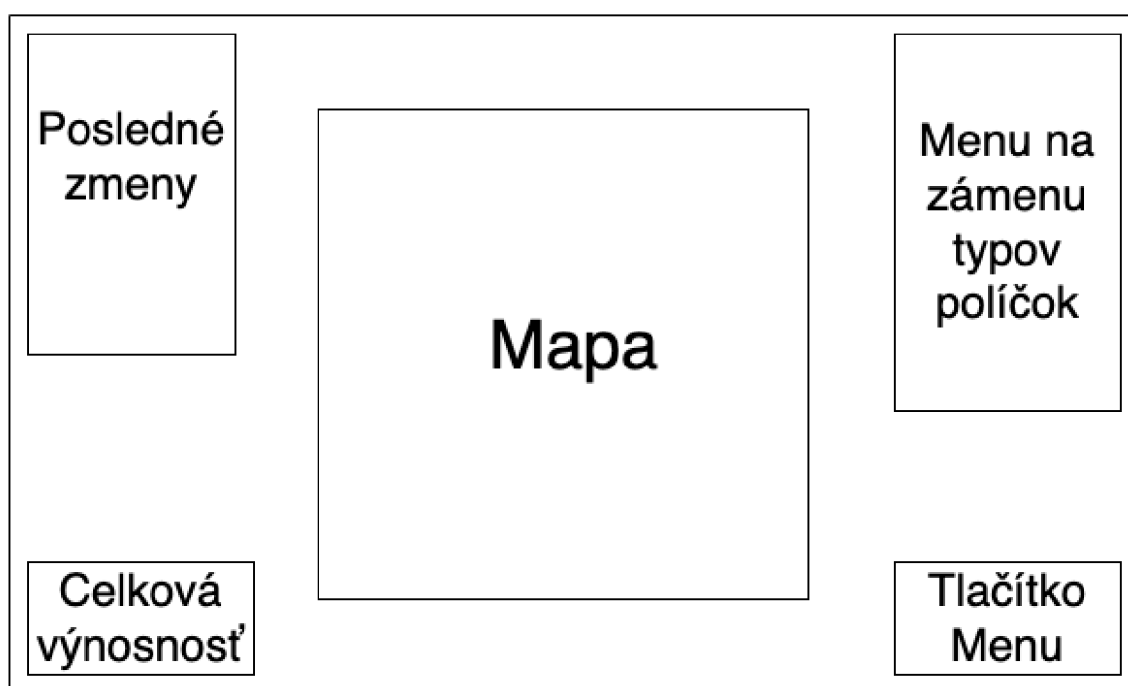
Na prácu s herným menu sa budú využívať tlačítka, ktoré otvárajú následne otvoria kontextové okno. Na tieto tlačítka je potrebné kliknúť ľavým tlačidlom myši. Hráč má vždy možnosť sa vrátiť do počiatočného menu, pomocou tlačítok, z akejkolvek scény.



Obr. 4.1: Návrh obrazovky herného menu.

Scéna pre mapu sa stáva aktívnou pri odoslaní požiadavku na vytvorenie alebo načítanie hry zo scény herného menu. Obsahuje mapu dediny, ktorá sa automaticky vytvára pri aktivovaní scény, všetky informácie o dedine a celú funkcionality. O funkcionality tejto scény sa budú starať samostatné skripty, ktoré budú rozdelené podľa funkcionality.

Hráč bude schopný si mapu posúvať pomocou pravého tlačidla na myši a približovať pomocou kolieska na myši. Na jednotlivé políčka je možné kliknúť ľavým tlačidlom myši. Po kliknutí sa majú zobrazíť detailné informácie o políčku. Pre zobrazenie základných informácií stačí prejsť kurzorom myši po políčku a informácie sa automaticky zobrazujú na obrazovke. Na pravej strane obrazovky bude menu, kde je možnosť si zakliknúť typ políčka a následne meniť jednotlivé políčka na mape klikaním ľavého tlačidla myši zakliknutím typom. Pre zrušenie menenia políček bude potrebné kliknúť na tlačítko v pravom hornom rohu. V ľavom dolnom rohu sa bude nachádzať tlačítko, pomocou ktorého sa dá zmenená mapa uložiť, vypnúť hru alebo odísť do herného menu.



Obr. 4.2: Návrh obrazovky mapy.

Cieľ hry

Hranie hry pozostáva z vytvorenia mapy dediny pomocou textového súboru. Na mape bude následne možnosť meniť jednotlivé políčka a pozorovať vplyv týchto zmien na celkovú výnosnosť dediny a index vody pri políčkach. Nakoniec bude možné si uložiť už upravenú mapu znova do textového súboru pre jej budúce alebo opätovné načítanie.

Cieľom hry bude správne simulovať výnosnosť dediny a zádržnosť vody v dedine pri editácii vytvorenej mapy dediny. Hráč bude mať možnosť zisťovať vplyv menenia políček na celkovú výnosnosť dediny alebo výnosnosť jednotlivých políček. Na základe zistení bude vedieť určiť aké okolnosti vplývajú na výnosnosť a čo je pre mapu dediny prospešné.

4.3 Herné časti a mechaniky

Na zjednodušenie práce som rozdelil celú hru na jednotlivé časti, ktoré budú postupne vytvárané a spájané do jedného celku. Každá časť spĺňa jednu alebo viac herných mechaník.

Vytvorenie mapy

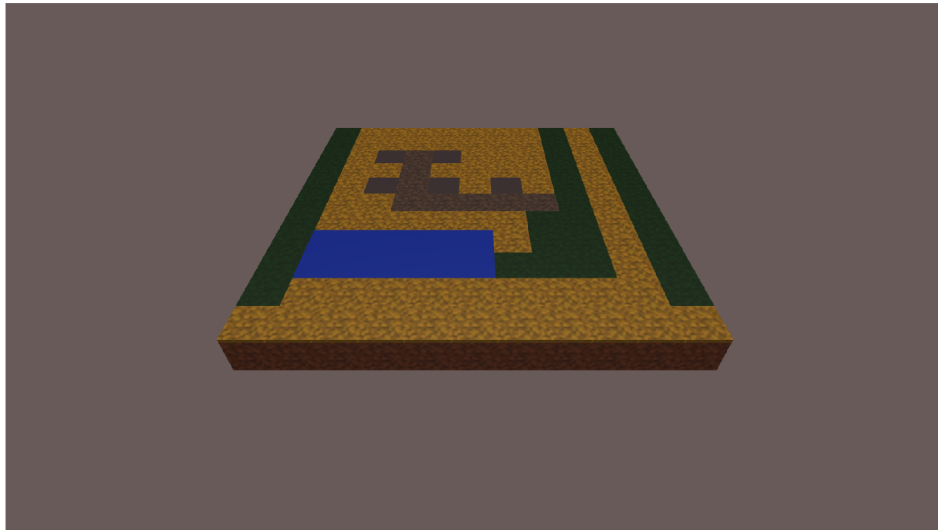
Hráč má možnosť vytvoriť mapu dediny tromi spôsobmi. Prvý spôsob je pomocou daného textového súboru, ktorý si hráč upraví sám podľa svojich predstáv vytváratej dediny. Druhý spôsob sa realizuje už pomocou hrania hry, kde je hráč schopný textový súbor z prvého spôsobu upraviť pomocou vstupného textového poľa. Pre obidva tieto spôsoby platí rovnaká syntax, pomocou ktorej sa definuje veľkosť mapy a typ jednotlivých políčok. Tretím spôsobom je načítanie už predom hranej a hráčom uloženej hry.

Uloženie mapy

Mapa je uložená v štruktúre, ktorá obsahuje jednotlivé políčka a ich základné vlastnosti. Medzi základné vlastnosti patrí typ políčka, jeho základný index výnosnosti a posledná zmena indexu výnosnosti pri editovaní mapy. Existuje 5 rôznych typov políčok, ktoré sú les, rieka, pole, cesta a dom. Každý typ má iný základný index. Ich hodnoty sú radené v poradí $pole > les \geq dom > rieka > cesta$.

Zloženie mapy

Mapa je vytvorená pomocou kockovanej mriežky, do ktorej sú vkladané jednotlivé typy políčok. Každý typ políčka má svoju charakteristiku a informácie o tom, ako ovplyvňuje celkový výnos mapy dediny a zádržnosť vody. Pre každý typ políčka je vytvorený jeden prefabrikát. Pomocou týchto prefabrikátov sa zaplní kockovaná mriežka o veľkosti, ktorú zadá hráč pomocou textového súboru alebo vstupného textového poľa.



Obr. 4.3: Základný návrh mapy dediny na kockovanej mriežke 11x11.

Kontrola validity mapy

Pri editovaní mapy sú určené základné podmienky, ktoré by mala mapa dodržiavať. Tieto podmienky slúžia k tomu, aby bol užívateľ upozornený pri vytváraní nezmyselnej dediny. Pri porušení podmienky pri zmenení políčka vyskočí červené vyskakovacie okno, ktoré obsahuje chybovú hlášku. Ak mapa nie je validná, má to razantný dopad na celkovú výnosnosť dediny. Pri odstránení porušenej podmienky sa okno s chybovou hláškou skryje a výnosnosť dediny sa vráti do normálu.

Základné podmienky pre vytvorenie validnej mapy sú:

- **Veľkosť mapy** - Mapa musí mať súradnice v rozsahu 5x5 až 15x15.
- **Zastúpenosť políček** - Zastúpenosť lesov a polí na mape musí byť väčšia ako 1/7 veľkosti mapy. Zastúpenosť riek musí byť väčšia ako 1/6 veľkosti mapy a na mape musia byť postavené minimálne 3 domy.
- **Cesta** - Pri každom dome musí byť cesta a zároveň musí existovať cesta von z dediny. Cesta ktorá nemá okolo seba žiadnu inú cestu sa berie taktiež za porušenie podmienky.

Ak nie je splnená aspoň jedna podmienka, od celkovej výnosnosti dediny sa odráta 40.

Ukazovanie informácií

Pre informovanie hráča o výnosnosti dediny a jednotlivých políček existujú rôzne spôsoby. Hráč bude mať na obrazovke neustále informáciu o celkovej výnosnosti dediny, pretože tá je najdôležitejšia. Základné informácie políčka sú k dispozícii pomocou hoover tabuľky, ktorá sa zobrazuje pri prechádzaní nad jednotlivými políčkami. Detailné informácie sú k dispozícii po kliknutí na jednotlivé políčka. Detailné informácie obsahujú všetky hodnoty políčka, ktoré sa využívajú pri výpočtoch výnosnosti.

Uchovávanie posledných zmien

Pre hráča sú k dispozícii informácie o posledných zmenách výnosnosti pri editovaní políček mapy dediny. Posledných 5 zmien celkovej výnosnosti dediny je možné vidieť na ľavej strane hry. Zmena výnosnosti sa taktiež uchováva pri jednotlivých políčkach. Jednotlivé informácie o zmene výnosnosti sa upravujú každým zmenením políčka na mape a majú farebné označenie na základe kladnej alebo zápornej zmeny.

Editácia mapy

Editovanie mapy je umožnené pomocou panelu, ktorý sa nachádza na pravej strane hry. Hráč pri prejdení do módu editovania mapy si bude môcť vybrať typ políčka a následne klikaním ľavého tlačidla na myši bude môcť zamieňať jednotlivé políčka na mape vybraným typom. Ukončenie módu editovania môže stlačením tlačítka v pravom hornom rohu.

Pauza a odchod do počiatočného menu alebo vypnutie hry

Hráč si môže pozastaviť hru pomocou stlačenia klávesy pre pozastavenie hry alebo tlačítka, ktoré sa nachádza v pravom dolnom rohu. Pri pozastavení hry sa objavia hráčovi možnosti ako sú vypnutie hry, odchod do počiatočného herného menu alebo uloženie hry. Ak nechce využiť ani jednu z týchto možností môže sa vrátiť naspäť do hry pomocou opätovného stlačenia klávesy pre pozastavenie hry alebo pomocou tlačítka pre pokračovanie v hre.

Počítanie výnosnosti políček

Výpočet výnosnosti jednotlivých políček mapy je najdôležitejšou úlohou tejto práce. Množstvo okolností vplýva na túto hodnotu. Medzi základné ovplyvňovatele, podľa ktorých sa vypočítava výnosnosť jednotlivých políček sú:

- **Početnosť vody, lesov, polí a domov v dedine** - Ideálnym výskyt domov na mape je 8% z celkovej mapy, v prípade lesov a polí je to 20% a riek 15%. Tieto počty sú zaokrúhlené nadol. Ak je dosiahnutý presný ideálny počet tak základný index typu políčka sa nemení. Ak je ideálny index nedosiahnutý alebo presiahnutý tak základná výnosnosť typu políčka sa lineárne mení podľa vzorca:

$$y = y - \frac{x * z}{10000} \quad (4.1)$$

kde premenná y znamená základný index políčka, premenná x znamená počet políček na mape a premenná z znamená o aký počet sa líši aktuálny počet políček od ideálneho.

- **Typ políčka** - Každý typ políčka má počiatočnú výnosnosť inú. V prípade ciest je najmenšia a v prípade polí najväčšia.

- Pole - 0,5.
- Rieka - 0,3.
- Les - 0,4.
- Cesta - 0,1.
- Dom - 0,4.

- **Cesta ku najbližšej vode a domu** - Pre každé políčko je vypočítaná vzdialenosť od rieky a od domu. Na základe týchto vzdialeností klesá alebo stúpa výnosnosť políčka. Výnosnosť stúpa ak je políčko čo najbližšie pri rieke a dome. Výnosnosť konkrétneho políčka sa mení podľa vzorca:

$$y = y + (3 - x) * 0,125 + (3 - z) * 0,075 \quad (4.2)$$

kde premenná y znamená výnosnosť konkrétneho políčka, premenná x znamená vzdialenosť od rieky a premenná z znamená vzdialenosť od domu. Maximálna hodnota premenných x a z je 3.

- **Kompaktnosť a vyváženosť lesov, riek a polí** - Výnosnosť políčka stúpa s počtom rovnakých typov políček v jeho okolí podľa vzorca

$$y = y + \frac{x}{50} \quad (4.3)$$

kde premenná y znamená výnosnosť konkrétneho políčka a premenná x znamená počet rovnakých typov políčka v okolí.

Nakoniec sa výnosnosť jednotlivých políček po všetkých výpočtoch upraví podľa veľkosti mapy pomocou vzorca:

$$y = \frac{100 * y}{x} \quad (4.4)$$

kde premenná y znamená výnosnosť konkrétneho políčka a premenná x počet políček na mape. Spočítanie všetkých výnosností jednotlivých políček vedie ku dosiahnutiu hodnoty v rozmedzí 0 - 100. Táto hodnota udáva celkovú výnosnosť dediny.

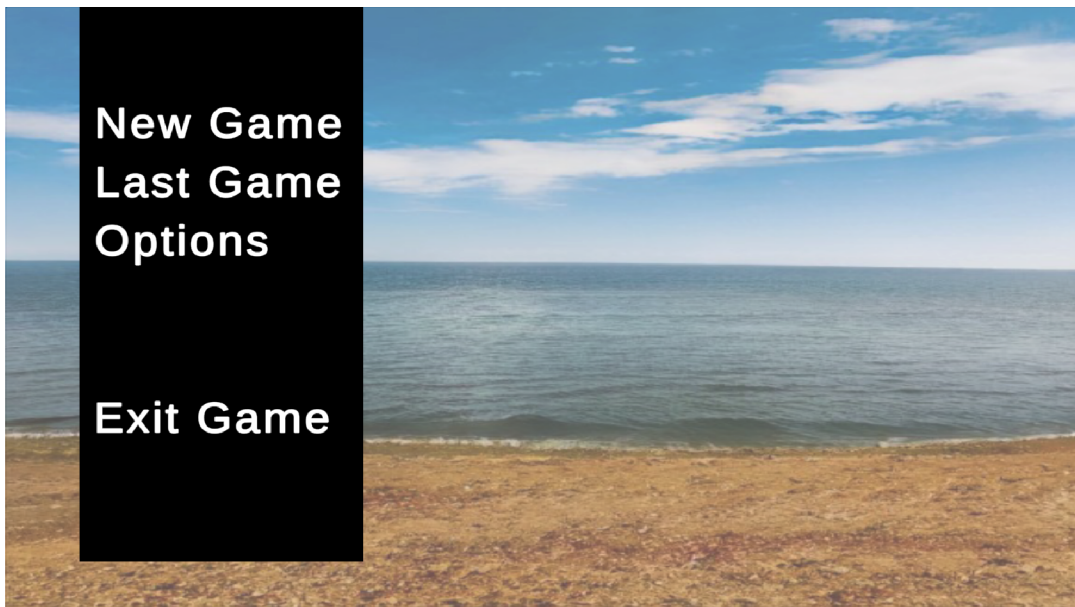
Kapitola 5

Implementácia herného menu

V tejto kapitole je detailne popísaná implementácia herného menu a popis jednotlivých herných objektov v tejto scéne. V tejto kapitole sú ďalej popísané jednotlivé skripty, ktoré pracujú herným menu.

5.1 Základ herného menu

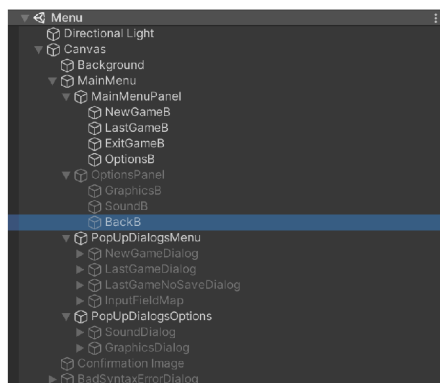
Celé herné menu je vytvorené v nezávislej scéne **Menu**. Táto scéna je 2D. Na implementáciu herného menu som využil ako základný prvok **canvas**. **Canvas** je herný objekt, do ktorého je možné vkladať rôzne UI elementy, ako sú napríklad tlačítka, rôzne texty, objekty a množstvo ďalších prvkov. **Canvas** je možné nastaviť na veľkosť hráčovej obrazovky, to znamená že veľkosť herného okna sa vždy nastaví podľa parametrov hráčovho monitora. **Canvas** môže navyše splniť úlohu obrázka v pozadí. Na obrázku 6.1 je možné vidieť canvas v ktorom je vložený obrázok mora, vytvorený pomocou programu **NVIDIA Canvas**.



Obr. 5.1: Obrázok herného menu hneď po spustení hry.

Objekty v hernom menu

Herné menu sa skladá z obrázkov, tlačítok, textov, začiarkavacieho políčka, výberu z možností a posuvného tlačítka. Na obrázku 5.2 je možné vidieť štruktúru Menu scény v Unity.



Obr. 5.2: Obrázok štruktúry scény Menu.

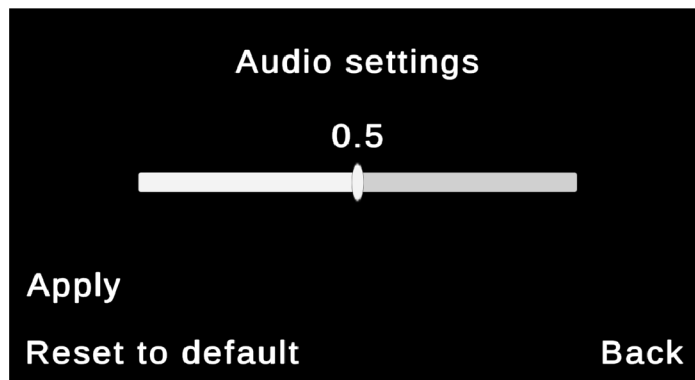
Obrázky

Obrázok je základ pri počiatocnom menu a taktiež všetkých vyskakovacích okien. Tieto obrázky majú všetky čisto čiernu farbu a rôzne veľkosti a umiestnenie. Do týchto obrázkov sú vložené jednotlivé herné objekty, s ktorými sa dá ďalej narábať. Na ľavej strane obrázku 5.1 sa nachádza tento čisto čierny obrázok v ktorom sú vložené tlačítka, v podobe bieleho textu.

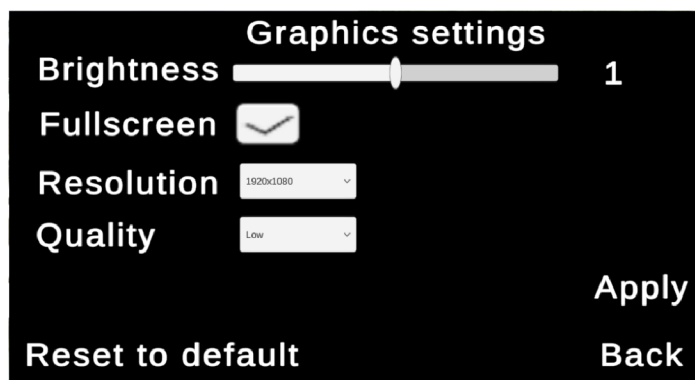
Tlačítka

V hernom menu sa vyskytuje množstvo tlačítok. Každé tlačítko má svoju osobitnú funkciu ktorú spĺňa. Základné štyri tlačítka, ktoré je možné vidieť hneď po spustení hry sú:

- **New Game** - Po kliknutí na toto tlačítko sa objaví vyskakovacie okno v ktorom je potrebné si vybrať spôsob, akým sa bude vytvárať základná mapa dediny. Hráč túto mapu bude môcť následne editovať v ďalšej scéne.
- **Last Game** - Ak má hráč uloženú skôr rozohranú hru, môže pokračovať v tejto hre ďalej pomocou tlačítka **Load Game**. Po kliknutí tohto tlačítka sa objaví vyskakovacie okno v ktorom si hráč môže načítať uloženú hru.
- **Options** - Pomocou tlačítka **Options** môže hráč meniť nastavenia hlasitosti zvuku a rôznych grafických prvkov hry. Po kliknutí sa objaví vyskakovacie okno s možnosťou voľby, ktoré nastavenia chce hráč meniť. Hráč má na výber meniť nastavenia zvuku hry pomocou tlačítka **Audio** alebo meniť nastavenia grafiky hry pomocou tlačítka **Graphics**. Po kliknutí tlačítka **Audio** sa objaví vyskakovacie okno v ktorom je vložený UI prvok posuvné tlačítko. Pomocou tohto prvku môžeme nastaviť úroveň hlasitosti v hre. Vyskakovacie okno po kliknutí tlačítka **Graphics** sa skladá z posuvného tlačítka, zaškrťavacieho políčka a dvoch výberov z možností. Pomocou týchto prvkov je možné meniť jas, kvalitu, rezolúciu hry alebo režim na celú obrazovku.



Obr. 5.3: Obrázok vyskakovacieho okna po kliknutí na tlačítko Audio.



Obr. 5.4: Obrázok vyskakovacieho okna po kliknutí na tlačítko Graphics.

Pri oboch vyskakovacích oknách máme taktiež tlačítko **Apply**, ktoré sa stará o ukladanie nastavení a tlačítko **Reset to default**, ktoré sa stará o resetovanie do pôvodných nastavení.

- Exit Game - Ak si hráč praje ukončiť hru, tak to môže zaistiť týmto tlačítkom.

V každom vyskakovacom okne je navyše tlačítko, ktoré po kliknutí vráti hráča do základného menu. Je tam z dôvodu, aby sa mal hráč ako vrátiť do základného menu pri chybnom kliknutí alebo iných okolnosti.

5.2 Skript na ovládanie herného menu

Skript, ktorý ovláda herné menu som pomenoval **MenuBehaviour**. V tomto skripte sa nachádzajú rôzne funkcie, ktoré majú za úlohu správne zobrazovať a skrývať jednotlivé vyskakovacie okná. Taktiež sa tu nachádzajú funkcie na správne uloženie nastavení hlasitosti a grafických prvkov. Pre správne fungovanie skriptu je potrebné mať pridelené pri scéne **Menu** v skripte **MenuBehaviour** všetky potrebné vyskakovacie okná a UI prvky, s ktorými následne chcem pracovať pomocou **C#** kódu.

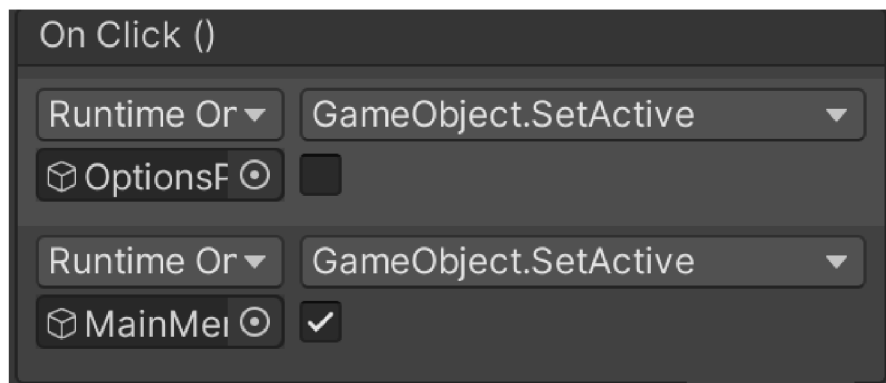
Menenie nastavení hry

Pri menení nastavení využívam balíček **TMpro**, ktorý mi dáva viac možností pri práci s UI prvkami. Pomocou **TMpro** balíčku sa viem dostať k hodnotám ktoré si zvolí hráč a následne s nimi pracovať a uložiť. Medzi často využívané objekty **TMpro** balíčku patria:

- **TMPro.Text** - Pomocou tohto objektu sa dá pristupovať k textovej zložke pomocou `.text`, k farbe pomocou `.color` a k fontu pomocou `.font`.
- **TMPro.Dropdown** - Tento objekt uchováva výber z možností, kde sa dajú získať alebo nastaviť hodnotu pomocou `.value`.
- **TMPro.Inputfield** - Tento objekt uchováva textové pole, do ktorého môže hráč vpisovať a pomocou `.text` sa dajú získať dáta ktoré napísal.

Skrývanie a zobrazovanie vyskakovacích okien

Na prácu so zobrazovaním a skrývaním objektov v unity existuje množstvo dostupných funkcií a spôsobov. Skrývanie a zobrazovanie objektov som riešil pomocou unity editora pri komponente **button**, kde som nastavil, ktoré okná majú byť skryté a ktoré zase zobrazené po stisnutí konkrétneho tlačítka.



Obr. 5.5: Obrázok nastavenia button komponentu.

Kapitola 6

Implementácia mapy dediny

V tejto kapitole je detailne popísaná implementácia mapy dediny a popis jednotlivých herných objektov v tejto scéne. V tejto kapitole sú ďalej popísané jednotlivé skripty, ktoré pracujú mapou dediny.

6.1 Zloženie scény

Na vytváranie máp som si na začiatku vytvoril prefabrikáty pre jednotlivé typy políčov, ktoré sa skladajú z dvoch častí:

- **Hlina - spodok** - Táto časť je vytvorená pomocou objektu `Cube` na ktorú je nanosená textúra hlíny, vytvorená v programe Blender.
- **Typ políčka - vrch** - Taktiež vytvorená pomocou objektu `Cube`, ktorý je navyše zmenšený a položený na spodnú časť. Každý vrch políčka má v sebe komponent `button` a preto je naňho v hre možné kliknúť myšou.

Existuje celkovo päť prefabrikátov pre jednotlivé druhy políčov. Tieto prefabrikáty sa líšia iba vo farbe vrchnej časti.



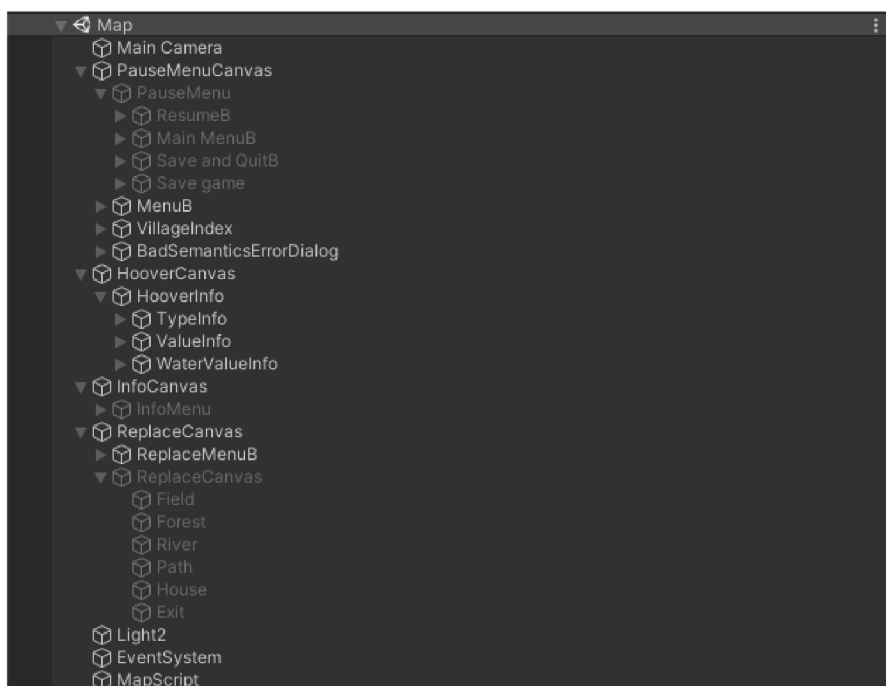
Obr. 6.1: Obrázok vyžívaných prefabrikátov.

Scéna `Map` sa skladá z týchto herných objektov:

- **Obrázky** - Obrázky sú využívané ako základ vyskakovacích okien rovnako ako v scéne `Menu`. Tieto obrázky sú čierne ale zároveň je ich transparentnosť nastavená tak, aby cez vyskakovacie okno bolo vidieť mapu v pozadí.
- **Textové polia** - Textové polia sa využívajú na odovzdávanie informácií hráčovi a zároveň získavanie informácií z mapy pre potrebné výpočty.
- **Tlačítka** - Pomocou tlačítok je hráč schopný otvárať vyskakovacie okná a pomocou nich vykonávať zmeny políčov alebo zisťovanie informácií. Každý herný objekt môže byť tlačítkom ak je k nemu priradená komponenta `button`.

- **Kamera** - Každá scéna potrebuje minimálne jednu kameru. Pomocou nej sa ovláda pohľad hráča na hru a pomocou okna **inšpektor** je možné meniť jej pozíciu, rotáciu a množstvo iných vecí.
- **Svetlo** - Pomocou svetla sa ožarujú jednotlivé políčka na mape. Bez svetla by sme mali mapu neprehľadnú a tmavú. Jej vlastnosti sa menia taktiež ako u kamery, za pomoci okna **inšpektor**.

Na obrázku 6.2 je možné vidieť štruktúru scény Map v unity.



Obr. 6.2: Obrázok štruktúry scény Map.

Syntax textového súboru pre vytvorenie a uloženie mapy dediny

Na vytvorenie mapy sa používa textový súbor, ktorý je potrebné vytvoriť. Tento textový súbor sa skladá z dvoch častí:

- **Veľkosť mapy dediny** - nachádza sa na prvom riadku textového súboru a má tvar $X \times Y$, kde X je počet riadkov a Y je počet stĺpcov.
- **Zloženie mapy dediny** - Každý nový riadok pozostáva z políčok, ktoré sa nachádzajú na mape v príslušnom riadku. Jednotlivé políčka sú charakterizované číslami 1(pole), 2(les), 3(rieka), 4(cesta), 5(dom) a sú oddelované čiarkou.

Pri ukladaní pozmenenej mapy vzniká textový súbor s rovnakou syntaxou pomocou štruktúry `_mapTiles`.

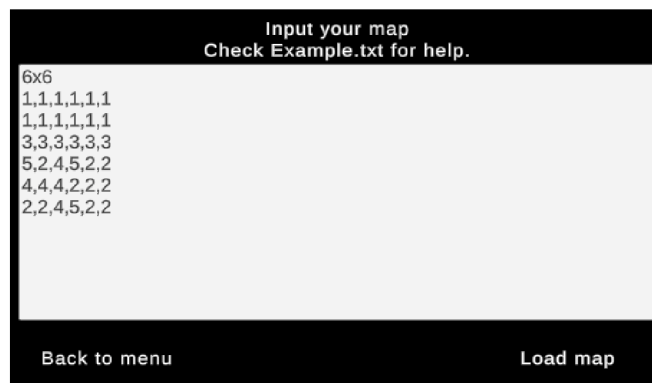
6.2 MapController skript

Tento skript je najrozsiahlejší a pomocou neho sa vytvára a ukladá mapa, počítajú jednotlivé výnosnosti políčok a celková výnosnosť, ukazujú informácie a menia políčka.

Vytváranie mapy

Vytváranie mapy sa odohráva prostredníctvom jednej z troch funkcií, ktorá sa vyberie podľa spôsobu vytvárania mapy. Spôsoby vytvárania mapy sú:

- **Novozačatá hra pomocou textového súboru** - V prípade ak sa hráč rozhodne vytvoriť hru pomocou textového súboru, volá sa funkcia `NewGameDialogCreate()`, ktorá kontroluje správnosť textového súboru pomocou `try` funkcie a pravidiel, ktoré sú potrebné pre správne vytvorenie mapy. V prípade chybného textového súboru sa hráčovi objaví vyskakovacie okno ktoré ho informuje o chybe. Ak v textovom súbore chyba nie je, načíta sa scéna `Map`.
- **Novozačatá hra pomocou vpisovacieho poľa** - Pri vybratí tohto spôsobu sa hráčovi zobrazí vpisovacie okno, do ktorého je potrebné napísať parametre mapy. Toto vpisovacie okno môžete vidieť na obrázku 6.3. Následne po napísaní mapy a odkliknutí tlačítka `Load map`, sa volá funkcia `NewGameInputMap()`, ktorá vpísaný text uloží do súboru `Map.txt` a zavolá funkciu `NewGameDialogCreate()`. Následne sa pokračuje ako pri prvom spôsobe.



Obr. 6.3: Obrázok vstupného poľa.

- **Uložená hra** - Pri vytváraní mapy pomocou uloženej hry sa očakáva prítomnosť súboru `MapSave.txt`, ktorý vznikol predošlým uložením hry hráčom. V prípade neexistencie tohto súboru, objaví sa vyskakovacie okno, ktoré informuje o neexistencii uloženej hry. Ak súbor existuje zavolá sa funkcia `LastGameDialog()`, ktorá následne načíta scénu `Map`.

Pri zavolaní scény `Map`, sa automaticky volá udalostná funkcia `Start()`. V tejto funkcii sa vytvára počiatočná mapa, pomocou inšancovania jednotlivých prefabrikátov na základe daného textového súboru. Pre každé políčko sa uloží do štruktúry `_mapTiles` typ políčka a priradia sa príslušné indexy. Typ políčka je získaný z daného textového súboru.

Ukladanie mapy

Uloženie mapy je dostupné pomocou tlačítok `Save game` a `Save and quit game`, ktoré sú dostupné v `Pause` menu. Po stlačení jedného z týchto tlačítok sa zavolá funkcia `MapSave()`, ktorá prekonvertuje štruktúru `_mapTiles` na textovú podobu a uloží ju do `MapSave.txt`.

Kontrola validity mapy

Kontrola podmienok prebieha prostredníctvom funkcie `MapCheck()`. Telo tejto funkcie prebieha v cykle cez všetky položky štruktúry `_mapTiles`. Pri každej položke zistí aký je to typ políčka pomocou `TileType`, na základe ktorého overí všetky podmienky. Pri zistení porušenia nejakej z podmienok zavolá funkciu `ConditionError(int type)`. Táto funkcia sa postará o zobrazenie chybovej hlášky pod oblasťou mapy, ktorá charakterizuje typ nespĺnenej podmienky a pozíciu políčka.

Funkcia `MapCheck()` sa volá pri každom zmenení políčka a pri vytvorení mapy. V prípade zistenia porušenia podmienky sa upraví výnosnosť dediny podľa vzorca:

$$y = y - 40 \quad (6.1)$$

kde y znamená výnosnosť dediny.

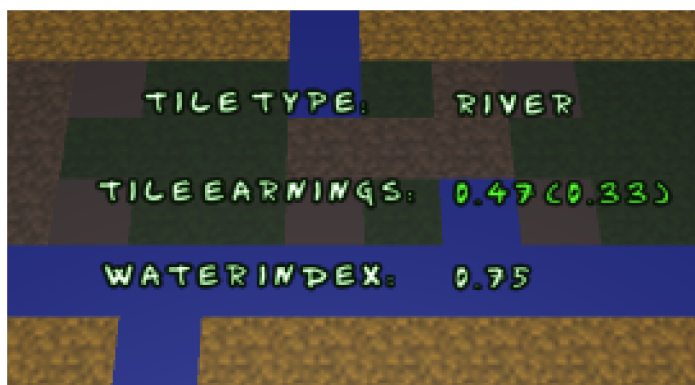
Informácia o výnosnosti dediny

Celková výnosnosť sa zobrazuje v ľavom dolnom rohu aplikácie. Výnosnosť sa zobrazuje červenou, zelenou a bielou farbou v závislosti zmeny výnosnosti.

Ukázanie informácií o políčku

Informácie o políčku sa získavajú za pomoci počítačovej myši. Zisťovanie, kde sa nachádza počítačová myš má za úlohu funkcia `RaySet()`, ktorá sa volá každý snímok v udalostnej funkcii `Update()`. V prípade ak sa hráč nachádza v `Pause menu`, funkcia `RaySet()` nie je volaná. Existujú dve možnosti ako získať informácie o jednotlivých políčkach:

- **Základné informácie** - Základné informácie o políčku je možné získať prejdением počítačovou myšou nad konkrétnym políčkom. Zobrazované informácie sú výnosnosť dediny, zádržnosť vody a typ políčka. V prípade výskytu počítačovej myši nad políčkom sa volá funkcia `HoverFunction()`. Funkcia `HoverFunction()` zaisťuje správne priradenie základných informácií a následne ich okamžité zobrazenie pri políčku.



Obr. 6.4: Obrázok základných informácií.

- **Detailné informácie** - Detailné informácie o políčku je možné získať kliknutím ľavého tlačítka na počítačovej myši nad konkrétnym políčkom. Zobrazované informácie sú výnosnosť dediny, zádržnosť vody, typ políčka, počet políčok od najbližšej rieky

a počet políček od najbližšieho domu. V prípade kliknutia ľavým tlačidlom na počítačovej myši nad konkrétnym políčkom sa volá funkcia `Pause(true)`. Táto funkcia má za úlohu správne priradenie detailných informácií a následne zobrazenie transparentného vyskakovacieho okna s detailnými informáciami. Transparentne okno je možné skryť pomocou tlačítka `Back to map` alebo stlačením klávesy `ESC`.



Obr. 6.5: Obrázok detailných informácií.

Vypočítavanie informácií o políčku

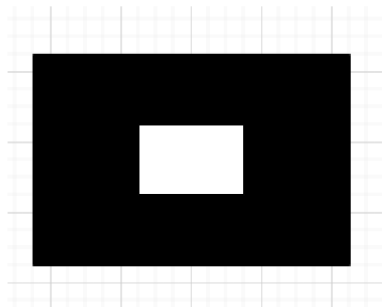
Pri výpočtoch informácií hrajú dôležitú rolu rôzne okolnosti na mape, ktoré sú opísané v kapitole 5. Pri implementácii som zistil chybu v návrhu. Chyba sa týkala nepresných výpočtov, ktoré viedli na popletenie hráča pri menení políček. Základné výnosnosti bolo potrebné upraviť nasledovne:

- Pole - 0,7.
- Rieka - 0,5.
- Les - 0,6.
- Cesta - 0,3.
- Dom - 0,6.

Rovnako bolo potrebné zmeniť ideálny počet v prípade lesov a polí na 30% a riek na 20%. Dôležitosť kompaktnosti a vyváženosti lesov, riek a polí bola pri návrhu veľmi nízka a z tohto dôvodu sa vzorec pre výpočet výnosnosti políčka vzhľadom na okolie upravil na:

$$y = y + \frac{x}{25} \quad (6.2)$$

Pri zisťovaní okolia políčka sa pozerá na políčka, ktoré sa nachádzajú na opísanej kružnici políčku. Na obrázku 6.6 je zobrazené okolie políčka čiernou farbu.



Obr. 6.6: Obrázok okolia políčka.

Vypočítanie výnosností majú za úlohu funkcie:

- **Closest(int type, int posi)** - Táto funkcia sa stará o zistenie najkratšej vzdialenosti políčka od domu a vody. Volá sa pomocou prvého parametru `int type`, ktorý určuje či hľadáme vzdialenosť k domu alebo rieke. Pomocou druhého parametru `int posi` sa určuje pozícia políčka v štruktúre `_mapTiles`, pre ktoré chceme túto vzdialenosť získať. Funkcia vracia `int`.
- **Calculate()** - Využíva funkciu `Closest(int type, int posi)` na získanie najkratšej vzdialenosti do políčka. Funkcia prechádza celou štruktúrou `_mapTiles` a vypočítava podľa vzorcov konkrétne výnosnosti pre políčka. Celková výnosnosť dediny sa počíta sumou všetkých konkrétnych výnosností. Táto funkcia sa volá pri každom zmenení políčka a pri vytvorení mapy.

Menenie políčk mapy

Menenie políčk prebieha za pomoci `Replace` menu. Tlačítko `Replace` menu na otvorenie tohto menu sa nachádza v pravom hornom rohu aplikácie. Po otvorení tohto menu sa naskytne pohľad na 5 tlačítok, ktoré reprezentujú jednotlivé typy políčk. Políčkom, ktoré sa vyberie, sa budú nahrádzať všetky políčka na ktoré sa klikne ľavým tlačidlom myši. Ak hráč už ďalej nechce nahrádzať klikne na tlačítko `EXIT` v mieste kde bolo tlačítko `Replace` menu.

Ukázanie posledných zmien

V ľavom hornom rohu sa nachádza päť textových polí, ktoré znázorňujú posledných päť zmien políčk hráčom. V prípade kladnej zmeny je text nastavený na zelenú farbu a v prípade zápornej na červenú farbu. Túto funkcionality má za úlohu funkcia `Calculate()`, ktorá zistí rozdiel medzi predošlou a aktuálnou výnosnosťou dediny a tú zapíše do prvého textového poľa. Následne ostatné hodnoty zapíše o riadok nižšie a poslednú zahodí.

Pause menu

`Pause` menu dáva možnosť hráčovi uložiť hru, odísť do herného menu alebo ukončiť hru pomocou príslušných tlačítok. Dostať sa do tohto menu je možné pomocou stlačenia klávesy `ESC` alebo pomocou tlačítka `Menu`, ktoré je umiestnené v pravom dolnom rohu aplikácie. Zatvoriť `Pause` menu je možné opätovným stlačením klávesy `ESC` alebo pomocou tlačítka `Resume`, ktoré sa nachádza na vrchole `Pause` menu.

V udalostnej funkcii `Update()` sa volá každý snímok funkcia `GamePausedFunc()`, ktorá kontroluje či nebolo stlačené tlačítko `ESC`. V prípade ak stlačené bolo a zároveň je zatvorené `Pause menu`, tak sa zavolá funkcia `Pause()`. Táto funkcia má za úlohu zobrazíť `Pause menu` a zároveň skryť nepodstatné objekty. Ak `Pause menu` už bolo otvorené a došlo k stlačeniu klávesy `ESC`, tak sa zavolá funkcia `Resume()`. Táto funkcia skryje `Pause menu` a zobrazí naspäť objekty, ktoré skryla funkcia `Pause()`.

Pri opustení `Pause menu` pomocou tlačítka `Resume` sa volá funkcia `Resume()` a pri stlačení tlačítka `Menu` pri hraní hry sa volá funkcia `Pause()`.



Obr. 6.7: Obrázok `Pause menu`.

6.3 `MapMouseMove` skript

Tento skript sa stará o správne narábanie s mapou, pomocou počítačovej myši. Za pomoci tohto skriptu vie hráč posúvať mapu z ľavej strany do pravej strany pomocou pravého tlačidla na počítačovej myši a približovať si mapu pomocou kolieska na počítačovej myši.

Celú túto funkcionálnu zabezpečuje funkcia `CameraMove()`, ktorá sa volá raz za snímok pomocou udalostnej funkcie `Update()`. Funkcia `CameraMove()` zisťuje či hráč ma stlačené pravé tlačidlo na počítačovej myši. Ak zistí že je stlačené, korektne pohybuje kamerou na základe hráčovho pohybu s počítačovou myšou. Približovanie mapy je riešené pomocou funkcie `Input.GetAxis("Mouse ScrollWheel")`, ktorá vracia polohu kolieska na počítačovej myši. Pomocou tejto polohy mením zorné kamery, čo má za následok približovanie alebo oddiaľovanie od mapy.

Kapitola 7

Testovanie

Testovanie aplikácie prebiehalo priebežne na v domovskom prostredí a na konzultáciách. Následné recenzie a pripomienky značne ovplyvnili hlavne vzhľad a funkcionálnosť aplikácie. Táto kapitola sa zameriava na zhodnotenie testovania počas vývoja a záverečné testovanie finálneho produktu.

Testovanie počas vývoja

Testovanie prebiehalo po každom významnom zmenení aplikácie. Hlavným dôvodom testovania bolo získanie spätnej väzby na užívateľské rozhranie a simuláciu výnosnosti.

Pomocou spätnej väzby sa neustále upravovalo užívateľské rozhranie, pre zjednodušenie editovania mapy. Prebiehalo taktiež spresňovanie výpočtov a ich dôležitosť.

Experiment na náročnosť vytvorenia mapy dediny

Cieľom tohto testovania bolo zistiť náročnosť vytvorenia mapy dediny a získať spätnú väzbu na možné zlepšenie do budúcnosti. Testovanie prebiehalo v domovskom prostredí na osobnom počítači pomocou piatich testerov.

Nasledujúca tabuľka udáva informácie, za aký čas sa podarilo testerovi vytvoriť mapu o veľkosti 10x10 políčok.

Meno testera	počet minút
Peter	3 minút
Viera	5 minút
Tomáš	X minút
Bianka	6 minút
Miloš	7 minút

Pri testerovi **Tomáš**, došlo k zlyhaniu pri vytváraní mapy, kvôli nepochopeniu syntaxe vytvárania. Vytváranie mapy sa zdalo testerom veľmi zložité a náročné. Toto testovanie odhalilo nedostatok vo vytváraní mapy a možnosť na budúce zlepšenie.

Experiment na logické zistenie okolností, ktoré vplývajú na výnosnosť

Cieľom testovania bolo zistiť presnosť výpočtov výnosnosti dediny a náročnosť zistenia všetkých okolností. Každý tester mal možnosť meniť políčka v predom vytvorenej dedine o veľkosti 10x10 a zisťovať dôvody zmien výnosnosti dediny. Úlohou testerov bolo zistiť všetky okolnosti, ktoré vplývajú na výnosnosť dediny, za čo najmenší počet zmien políčok.

Nasledujúca tabuľka udáva informácie o počte zmien políčok, po ktorých tester vedel opísať všetky okolnosti, ktoré vplyvajú na výnosnosť dediny.

Meno testera	počet zmien políčok
Peter	25
Viera	40
Tomáš	35
Bianka	32
Miloš	23

Okolnosti, ktoré vplyvajú na výnosnosť dediny, sa zdali testerom veľmi logické a presnosť výpočtov si chválili. Nebolo pre nich príliš ľahké zistiť všetky okolnosti ale zároveň ani príliš ťažké. Tento experiment dokázal splnenie konkrétneho cieľu tejto bakalárskej práce.

Experiment na rýchlosť zmeny mapy na určitú výnosnosť

Cieľom testovania bolo zistiť náročnosť vytvorenia dediny s určitou výnosnosťou. Každý tester dostal rovnakú mapu s určitou výnosnosťou a jeho úlohou bolo čo najrýchlejšie dosiahnuť výnosnosť 90.

Nasledujúca tabuľka udáva informácie o počte minút, po ktorých tester dosiahol výnosnosť dediny väčšiu ako 90. Počiatočná výnosnosť predom vytvorenej dediny bola 50 a veľkosť dediny bola 13x13.

Meno testera	počet minút
Peter	12 minút
Viera	16 minút
Tomáš	14 minút
Bianka	11 minút
Miloš	13 minút

Nasledujúca tabuľka udáva informácie o počte minút, po ktorých tester dosiahol výnosnosť dediny väčšiu ako 90. Počiatočná výnosnosť predom vytvorenej dediny bola 70 a veľkosť dediny bola 6x6.

Meno testera	počet minút
Peter	5 minút
Viera	7 minút
Tomáš	9 minút
Bianka	7 minút
Miloš	10 minút

Týmto experimentom bola overená správna obťažnosť hry. Počet minút na dosiahnutie ideálneho výsledku lineárne stúpala spoločne s veľkosťou mapy. Testerom nerobilo problém vytvoriť dedinu s výnosnosťou 90 a trvalo to každému približne rovnako. Možné zlepšenie do budúcnosti podľa testerov, je pridanie ďalších rôznych ovplyvňujúcich okolností.

Záverečné zhodnotenie

Pomocou experimentov na malej vzorke testerov sa zistili možné vylepšenia aplikácie v budúcnosti. Hlavným problémom aplikácie je náročné vytváranie mapy, ktoré by išlo urobiť jednoduchšie a intuitívnejšie. Pomocou experimentov sa otestoval a dokázal splnený konkrétny cieľ tejto bakalárskej práce.

Kapitola 8

Záver

Cieľom tejto bakalárskej práce bolo vytvoriť edukatívnu hru v hernom engine Unity. Hráč má možnosť si vytvoriť mapu dediny nad ktorou následne zamieňa políčka za iné typy. Počas menenia políčok sa učí o okolnostiach, ktoré vplývajú na výnosnosť dediny a zádržnosti vody v krajine. Na začiatku bol vysvetlený pojem počítačová hra, ako sa rozdeľujú počítačové hry, opis edukatívnych hier a vysvetlené opatrenia proti vysychaniu krajiny. Bol vysvetlený proces plánovania a vytvárania hry. Následne bol popísaný herný engine Unity, jeho editor a porovnanie s Unreal engine. Konkrétnym cieľom tejto bakalárskej práce bolo vytvoriť hru, ktorá umožňuje vytvoriť a editovať mapu dediny. Na základe editovania mapy sa má neustále vypočítať výnosnosť dediny, pomocou podstatných okolností. Podľa týchto zmien má hráč možnosť učiť sa a zisťovať okolnosti, ktoré vyplývajú z výnosnosti dediny a jej neustáleho sa menenia. Pomocou testovania bol tento konkrétny cieľ overený.

Možností ako pokračovať v tejto bakalárskej práci je množstvo. Hráč by mal možnosť mať možnosť v ďalšej fázy hry vytvoriť si mapu dediny, prevádzať rôzne vylepšenia typické pre jednotlivé časti dediny a sledovať ich dopad pomocou grafických zobrazení a výsledných hodnôt zádržnosti vody a výnosnosti dediny. Neskôr sa ponúka kontaktovať meteorologickú stanicu za cieľom získania informácií, ktoré by sa následne využili na spresnenie počítania zádržnosti vody. Ďalšou možnosťou ako pokračovať by bolo pridať rôzne vrstvy zemín a opätovne spresniť počítanie zádržnosti vody.

Literatúra

- [1] *Unreal Engine vs Unity*. 2022. [Online; prístup 19. 01. 2022]. Dostupné z: <https://sk.education-wiki.com/1099355-unreal-engine-vs-unity>.
- [2] B., M. *Plánovanie je základom úspechu*. 2021. [Online; prístup 26. 04. 2022]. Dostupné z: <https://www.sashe.sk/IdeaTree/journal/planovanie-je-zakladom-uspechu>.
- [3] BENES. *Ako vytvoriť počítačové hry*. 2021. [Online; prístup 26. 04. 2022]. Dostupné z: <https://abcslovakia.sk/ako-vytvori-poitaove-hry-11-krokov/>.
- [4] ČERNÍK, J. *Herní design*. Univerzita Pardubice. 2018.
- [5] CSATÁRYOVÁ, M., ŠEBEŇ, V. a ŠECHNÝ, M. *Virtuálne laboratórium v astronómii—programy Stellarium a Aladin. Tvorivý učiteľ—Festival Fyziky*. 2011.
- [6] DOSTÁL, J. *Instructional software and computer games—tools of modern education. Journal of Technology and Information Education*. Citeseer. 2009, zv. 1, č. 1, s. 23.
- [7] HRDINKA, T., NESLÁDKOVÁ, M., DAVIDOVÁ, T., PUNČOCHÁŘ, P. et al. *Příprava a zpracování Koncepce na ochranu před následky sucha pro území České republiky. Vodohospodářské technicko-ekonomické informace. Výzkumný ústav vodohospodářský TG Masaryka, veřejná výzkumná instituce*. 2017, zv. 59, č. 4.
- [8] RYLICH, J. *Počítačové hry jako fenomén nových médií*. Univerzita Karlova, Filozofická fakulta. 2011.
- [9] SINGH, H. *Unreal Engine vs. Unity*. 2021. [Online; prístup 25. 04. 2022]. Dostupné z: https://theninehertz.com/blog/unreal-engine-vs-unity-3d-games-development#Conclusion_Unreal_Engine_vs_Unity_3D_Games_Development.
- [10] ŠMÍD, A. *Comparison of unity and unreal engine. Czech Technical University in Prague*. 2017.
- [11] ŠPAČKOVÁ, A. *Možnosti využití recyklace vody*. 2018. B.S. thesis. České vysoké učení technické v Praze. Vypočetní a informační centrum.
- [12] UNITY TECHNOLOGIES. *Order of execution for event functions*. 2022. [Online; prístup 25. 04. 2022]. Dostupné z: <https://docs.unity3d.com/Manual/ExecutionOrder.html>.
- [13] UNITY'S INTERFACE. *Unity User Manual*. 2022. [Online; prístup 25. 04. 2022]. Dostupné z: <https://docs.unity3d.com/Manual/UsingTheEditor.html>.

- [14] WIKIPEDIA CONTRIBUTORS. *Počítačová hra* — *Wikipédia: Slobodná encyklopédia*. 2021. [Online; prístup 17. 01. 2022]. Dostupné z: https://sk.wikipedia.org/w/index.php?title=Po%C4%8D%C3%ADta%C4%8Dov%C3%A1_hra&oldid=7193339.
- [15] WIKIPEDIA CONTRIBUTORS. *Spacewar!* — *Wikipedie: Slobodná encyklopedie*. 2021. [Online; prístup 17. 01. 2022]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Spacewar!&oldid=20091345>.
- [16] WIKIPEDIA CONTRIBUTORS. *Unity (herní engine)* — *Wikipedie: Otevřená encyklopedie*. 2021. [Online; prístup 22. 04. 2022]. Dostupné z: [https://cs.wikipedia.org/w/index.php?title=Unity_\(hern%C3%AD_engine\)&oldid=19960664](https://cs.wikipedia.org/w/index.php?title=Unity_(hern%C3%AD_engine)&oldid=19960664).
- [17] ZAREBA, M. *Unity vs Unreal comparison*. 2021. [Online; prístup 25. 04. 2022]. Dostupné z: <https://4experience.co/unity-vs-unreal-comparison-choose-the-best-engine-for-you/>.

Príloha A

Obsah priloženého pamäťového média

- **source** - Adresár so všetkými zdrojovými súbormi, ktoré sú potrebné pre vytvorenie aplikácie.
- **BP-latex** - Adresár so zdrojovými súbormi technickej správy.
- **BP-xolear00.pdf** - PDF súbor s textovou časťou bakalárskej práce.
- **README.md** - Návod na vytvorenie aplikácie a jej spustenie.
- **video.mp4** - Demonstračné krátke video z hrania vytvorenej edukatívnej hry.
- **plagat.png** - Plagát vytvorenej edukatívnej hry.