

**STREDNÁ PRIEMYSELNÁ ŠKOLA JOZEFA MURGAŠA
BANSKÁ BYSTRICA**



3D TOWER DEFENSE HRA
MATURITNÁ PRÁCA PČOZ forma b)

PETER KAPSIAR, IV. C
BANSKÁ BYSTRICA 2022/2023

**STREDNÁ PRIEMYSELNÁ ŠKOLA JOZEFA MURGAŠA,
HURBANOVA 6, BANSKÁ BYSTRICA**

Meno a priezvisko:	Peter Kapsiar
Trieda:	IV. C
Školský rok:	2022/2023
Odbor:	2675 M elektrotechnika
Názov práce:	3D tower defense
Zadanie práce:	Navrhnuť 3D hru v unity Vytvoriť 3D modely pre hru Vytvoriť assety a textúry pre hru Naprogramovať logiku pre jednotlivé prvky Naprogramovať systém pre pokročenie v hre Overiť hrateľnosť Napísať sprievodnú dokumentáciu
Dátum zadania:	24. 10. 2020
Konzultant:	Ing. Jozef Dokoupil
Podpis konzultanta:

ČESTNÉ VYHLÁSENIE

Čestne vyhlasujem, že túto maturitnú prácu som vypracoval samostatne s použitím uvedených literárnych zdrojov. Som si vedomý/vedomá dôsledkov, ak uvedené údaje nie sú pravdivé.

Banská Bystrica 24. 2. 2023

.....

vlastnoručný podpis autora

Obsah

ZOZNAM OBRÁZKOV, TABULIEK A GRAFOV	5
Zoznam obrázkov.....	5
ÚVOD	6
1 TEORETICKÉ VÝCHODISKÁ	7
2 CIELE PRÁCE.....	10
2.1 Funkčnosť	10
2.2 Hratel'nosť	10
2.3 Vizualna stránka	10
2.4 Rozširiteľnosť	10
3 MATERIÁL A METODIKA PRÁCE.....	12
3.1 Myšlienka	12
3.2 Informácie	12
3.3 Tvorba prostredia	13
3.4 Menu.....	14
3.5 Hlavný ovládač	14
3.6 Nepriatelia.....	15
3.7 Spawner.....	16
3.8 Veže.....	17
3.9 Upgrade systém	18
3.10 Projektily	19
3.11 Užívateľské rozhranie	19
4 VÝSLEDKY PRÁCE A DISKUSIA.....	21
5 ZÁVER PRÁCE	22
ZHRNUTIE	23
ZOZNAM POUŽITEJ LITERATÚRY	24

ZOZNAM PRÍLOH.....	25
PRÍLOHY	26
1. Ovládanie.....	26

ZOZNAM OBRÁZKOV, TABULIEK A GRAFOV

Zoznam obrázkov

Obrázok 1 Unity logo.....	12
Obrázok 2 Hracia plocha.....	13
Obrázok 3 Menu.....	14
Obrázok 4 Nepriatelia	15
Obrázok 5 Začiatočný bod	16
Obrázok 6 Kanón	17
Obrázok 7 Katapult	17
Obrázok 8 Obchod veží.....	18
Obrázok 9 Upgrade menu	18
Obrázok 10 Bomba	19
Obrázok 11 Náboj	19
Obrázok 12 Ovládací panel.....	19

ÚVOD

V súčasnej dobe sú 3D hry stále populárnejšie a stávajú sa dôležitou súčasťou herného priemyslu. Vývoj 3D hier je však často komplikovaný proces, ktorý vyžaduje značné množstvo času a úsilia. Preto je nevyhnutné použiť efektívne nástroje a technológie na uľahčenie tohto procesu.

Pre výber tejto témy sme sa rozhodli kvôli dlhoročným skúsenostiam a osobného záujmu o programovanie. Za celú dobu čo programujeme sme prešli cez množstvo oblastí kde sa dá využiť programovanie. Od jednoduchých algoritmov, cez konzolové aplikácie, až po programovanie robotov či arduina. Preto si myslíme že tento projekt je ideálna príležitosť vyskúšať si a naučiť sa niečo nové o programovaní 3D hier v Unity.

Cieľom tejto práce je vytvoriť funkčnú a vizuálne príjemnú 3D hru v Unity. Na tvorbu modelov a dizajnu levelu použijeme Unity ProBuilder ktorý mi umožní jednoducho a rýchlo vytvoriť základné postavy, predmety a okolie. Aby hra fungovala pre jednotlivé funkčné objekty naprogramujeme skripty v jazyku C# vo vývojovom prostredí Microsoft Visual studio 2022.

1 TEORETICKÉ VÝCHODISKÁ

Unity je viacrozmerné vývojové prostredie a herný engine, ktorý umožňuje programátorom, výtvarníkom a dizajnérom vytvárať a publikovať rôzne druhy interaktívneho obsahu, ako sú napríklad počítačové hry, simulácie a vizualizácie. Unity umožňuje vytvárať aplikácie pre rôzne platformy, vrátane desktopových počítačov, mobilných zariadení, herných konzol a virtuálnej reality. Unity ponúka aj možnosť tvorby založenej na princípe vizuálneho programovania, čo znamená, že vývojári môžu tvoriť funkcie, scény a interakcie pomocou grafického rozhrania bez nutnosti písania zdrojového kódu. Unity tiež ponúka širokú paletu nástrojov pre optimalizáciu výkonu, testovanie a správu projektov. Tento engine sa často používa na tvorbu hier, ale môže byť tiež použitý na vytváranie mnohých iných interaktívnych aplikácií.

Game engine (herný engine) je softvérový nástroj, ktorý programátori používajú na vývoj a tvorbu videohier. Jedná sa o komplexný systém, ktorý obsahuje množstvo rôznych funkcií a nástrojov, ktoré zjednodušujú vývoj hier. Game engine umožňuje vývojárom pracovať so zdrojovými kódmi hry, vytvárať a editovať herné objekty, definovať ich vlastnosti a spravovať herné prostredie. Typické funkcie herného engine zahŕňajú správu grafiky, fyziky, zvukových efektov, umelú inteligenciu, správu používateľského rozhrania, sieťové funkcie a podporu rôznych platforiem a operačných systémov. Herný engine tak uľahčuje programátorom tvorbu hier, keďže nemusia vytvárať každú funkciu a nástroj sami.

Unity ProBuilder je integrovaný nástroj v hernom engine Unity, ktorý umožňuje vytvárať 3D modely priamo v editori Unity. Tento nástroj umožňuje vývojárom a dizajnérom vytvárať scény a objekty pomocou kombinácie primitívnych geometrických tvarov a intuitívnych nástrojov na úpravu vrcholov, hrán a plôch. Unity ProBuilder umožňuje vytvárať jednoduché aj komplexné geometrie, ktoré môžu byť použité na tvorbu detailných svetov a objektov v počítačových hrách a iných interaktívnych aplikáciách. Ďalšie funkcie nástroja zahŕňajú textúrovanie, osvetlenie, kolízie a exportovanie do rôznych formátov. ProBuilder je ideálnym nástrojom pre tvorcov hier, ktorí chcú vytvárať svety a objekty rýchlo a efektívne priamo v Unity bez nutnosti používať ďalšie externé aplikácie na modelovanie.

Unity Asset Store je online platforma, ktorú poskytuje spoločnosť Unity Technologies kde môžu vývojári hľadať a nakupovať digitálne obsahy a nástroje, ktoré potrebujú pre svoje projekty v Unity engine. Na Asset Store sú k dispozícii stovky tisíc 2D a 3D modelov, textúr, shaderov, scriptov, animácií, zvukových efektov a iných digitálnych prostriedkov, ktoré umožňujú vývojárom rýchlejšie a efektívnejšie vytvárať hry a aplikácie v Unity.

Microsoft Visual Studio je integrované vývojové prostredie (IDE) používané na vytváranie softvéru pre platformy Windows, Android, iOS, webové aplikácie a mnoho ďalších. Toto vývojové prostredie umožňuje programátorom písať, editovať a ladit zdrojové kódy v rôznych programovacích jazykoch, ako sú napríklad C#, C++, Visual Basic a Python. Microsoft Visual Studio poskytuje široké spektrum nástrojov a funkcií, ktoré umožňujú programátorom vytvárať sofistikované softvérové aplikácie, ako sú desktopové aplikácie, webové aplikácie, herné aplikácie a ďalšie. Toto vývojové prostredie tiež ponúka rôzne nástroje na testovanie, ladenie a optimalizáciu kódu, ktoré pomáhajú programátorom vytvárať kvalitný a bezpečný softvér. Microsoft Visual Studio je jedným z najpoužívanějších vývojových prostredí pre vývoj softvéru pre platformu Windows a iné platformy.

C# (vyslovuje sa ako "C šarp") je moderný vysokoúrovňový programovací jazyk, ktorý bol vyvinutý spoločnosťou Microsoft pre platformu .NET Framework. C# kombinuje prvky z jazykov C a C++, ale je navrhnutý tak, aby bol jednoduchší na používanie a bezpečnejší. C# sa často používa na vývoj desktopových aplikácií, webových aplikácií, mobilných aplikácií a hier pre platformu Windows, ale je tiež k dispozícii na iných platformách ako napríklad Linux a MacOS. C# je robustný a výkonný jazyk s modernými funkcionálnymi prvkami, ktoré umožňujú programátorom efektívne vytvárať kvalitné a bezpečné softvérové aplikácie.

Objektovo orientovaný programovací jazyk (OOP) je programovací jazyk, ktorý umožňuje tvorbu softvéru pomocou konceptov objektov. Objektová paradigma zahŕňa tvorbu tried, ktoré slúžia ako šablóny pre tvorbu objektov. Každý objekt má svoje

vlastnosti a funkcie, ktoré sú definované v rámci triedy, z ktorej bol vytvorený. OOP umožňuje vytvárať veľmi komplexné aplikácie pomocou jednoduchých objektov a tried, ktoré môžu spolupracovať a komunikovať medzi sebou. Medzi základné koncepty OOP patria dedičnosť, zapúzdzrenie a polymorfizmus. Dedičnosť umožňuje tvorbu nových tried na základe existujúcich tried. Zapúzdzrenie zabezpečuje, že dáta a funkcie sú skryté pred ostatnými časťami programu. Polymorfizmus umožňuje použiť jednu funkciu pre rôzne typy objektov. Použitie OOP prináša množstvo výhod, vrátane znovu použiteľnosti kódu, lepšej organizácie a prehľadnosti kódu a možnosti rozšírenia aplikácie bez jej zásadnej zmeny.

Tower defence hra je typ strategickej hry, ktorá sa zvyčajne hrá na počítači alebo na mobilnom zariadení. Hráč sa zvyčajne nachádza na nejakej obrazovke, na ktorej sa pohybujú nepriatelia. Jeho úlohou je brániť svoju základňu (alebo akýkoľvek iný cieľ) pred týmito nepriateľmi. Na tento účel hráč stavia rôzne obranné budovy, ako sú veže, ploty, ostne alebo rôzne ďalšie prekážky. Hráč môže vylepšiť svoje obranné stavby alebo kúpiť nové. Každá obranná budova má svoje vlastné vlastnosti, ako je rýchlosť streľby, presnosť alebo sila útoku, a hráč musí zvážiť, ktoré obranné budovy postaví a kde ich umiestni, aby čo najefektívnejšie ochránil svoju základňu. Cieľom hry je zvyčajne prežiť všetky úrovne, ktoré sa líšia v počte a sile nepriateľov a ich schopnostiach prekonať hráčove obrany.

2 CIELE PRÁCE

2.1 Funkčnosť

Pri tvorbe tejto hry sme sa zamerali hlavne na to aby boli všetky elementy hry plne funkčné a pracovali tak ako sme ich naplánovali. Dbali sme taktiež na to aby všetky prvky mali čo najmenej bugov a krajných situácií ktoré by mohli u používateľa ovplyvniť celkový dojem a zážitok pri hraní hry. Taktiež sme dbali veľký dôraz na optimalizáciu skriptov aby čo najmenej zaťažovali procesor na zariadení.

2.2 Hrateľnosť

Po dokončení základov a hlavnej štruktúry hry bolo dôležité vložiť čas do dôkladného testovania a doladovania hlavných herných mechaník aby bol priebeh hry a celkový zážitok pre hráča čo najzábavnejší a najpríjemnejší. Napríklad bolo dôležité dôkladne nastaviť ideálny počet životov nepriateľov, počet nepriateľov v jednotlivých vlnách, počet životov hráča, rýchlosť nepriateľov, ceny ochranných veží, rýchlosť strelby ochranných veží.

2.3 Vizuálna stránka

Aby hra vyzerala na prvý pohľad a počas hrania príťažlivo sme pridali niektoré zaujímavé a unikátne vizuálne prvky. Pre jednoliaty a intuitívny výzor sme väčšinu UI elementov vytvorili v trojrozmernom prevedení a umiestnili priamo do prostredia. Aby sme celkové prostredie trochu oživilí sme niektorým objektom pridali jednoduché animácie (napríklad nepriateľom). Na doplnenie niektorých animácií sme pri niektorých udalostiach k animáciám pridali aj jednoduché vizuálne efekty. Na dokončenie celkového výzoru sme celému prostrediu pridali minimálny post processing.

2.4 Rozšíriteľnosť

Cieľom práce bolo vytvoriť hru, ktorá by bola pripravená na možné budúce rozšírenie alebo úpravy, a ktorú by bolo možné ľahko upraviť a prispôbiť na základe potreby. Jedným z hlavných cieľov bolo vytvoriť čo najmodulárnejšiu architektúru, ktorá by umožnila pridávanie nových funkcií a elementov hry bez toho, aby sa to výrazne prejavilo na existujúcom kóde alebo celkovej funkčnosti hry. K tomuto účelu sme sa snažili použiť všeobecne uznávané návrhové vzory a programovacie paradigmy, ktoré uľahčujú rozširovanie softvéru. Okrem architektúry sme dbali aj na to, aby hra bola pripravená na možné grafické a zvukové rozšírenia, ako aj na možnosť pridávania nových

levelov, postáv a nepriateľov bez zásadných zmien v kóde. Všetky grafické prvky a zvuky boli preto organizované a spravované v samostatných balíčkoch, ktoré sa dali ľahko pridať alebo vymeniť.

3 MATERIÁL A METODIKA PRÁCE

3.1 Myšlienka



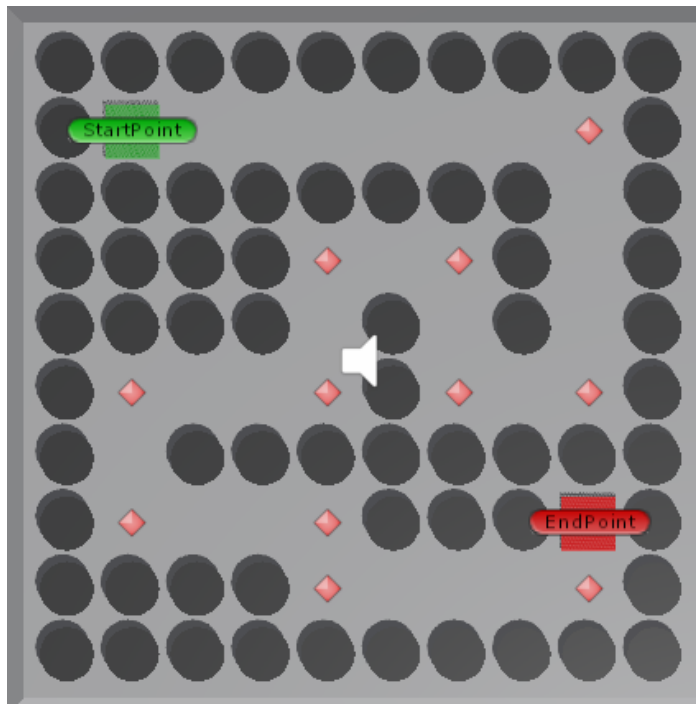
Obrázok 1 Unity logo

Pred tým než sme začali s tvorbou našej práce bolo veľmi dôležité si dopredu premyslieť ako bude finálny produkt vyzerat', zvoliť tému, štýl, žáner, zložitost' a taktiež aké bude obsahovat' prvky, herné mechaniky či postavy. Ako sme už spomínali v našom prípade sme sa rozhodli na základe predošlých skúseností zvoliť ako tému našej práce 3D hru vo vývojovom prostredí Unity ktoré nám umožňuje pomocou jednoduchých zabudovaných mechaník a nástrojov vytvárat' hry s takmer neobmedzenou komplexnosťou. Za žáner sme sa rozhodli vybrať tower deffence z toho dôvodu že jeho myšlienka je v základe jednoduchá ale jeho prevedenie môže byť uskutočnené stovkami spôsobov a taktiež je veľmi jednoduché k nemu pridať svoj vlastný zvrat.

3.2 Informácie

Po tom ako sme si dôkladne premysleli aký chceme dosiahnuť cieľový produkt bolo dôležité si na internete urobiť prieskum o type našej práce. Naštudovali sme si čo tvorí základ tower deffence hry, aké sú typické herné prvky, vylepšenia, aký sú typický nepriatelia, a bossovia. Taktiež sme sa venovali prieskumu existujúcich hier tohto typu, aby sme mohli zistiť, čo je v súčasnosti populárne a čo by sme mohli urobiť inak a pridať do našej hry niečo nové. Pozreli sme sa na rôzne herné mechaniky, ako napríklad možnosť upgradu obranných veží, rôzne druhy nepriateľov, možnosti budovania obrany a mnoho ďalšieho. V rámci prieskumu sme si taktiež preštudovali rôzne zdroje na tvorbu hier, ktoré nám poskytli užitočné tipy a triky pre prácu s Unity enginom. Okrem toho sme sa snažili pristupovať k tvorbe hry aj kreatívne a uvažovať mimo boxu, aby sme mohli pridať do hry niečo nové a zaujímavé.

3.3 Tvorba prostredia



Obrázok 2 Hracia plocha

Keď sme mali všetko premyslené začali sme s tvorbou prostredia. Typické prostredie tower defence hry väčšinou tvorí cesta po ktorej sa budú pohybovať nepriatelia. Na začiatku sme zadefinovali začiatočný bod kde sa budú objavovať noví nepriatelia, na cestu sme pridali smerové body po ktorých sa budú nepriatelia pohybovať a na koniec sme samozrejme zadefinovali koncový bod na ktorom nepriatelia môžu odoberať hráčovi životy. Okolo cesty sme pridali v mriežkovom usporiadaní platformy na ktoré bude hráč stavať svoje veže. Keďže sme samotné prostredie navrhli ako platformu rozhodli sme sa na prednú časť umiestniť prostredie pre užívateľské rozhranie - indikátor života, počítadlo peňazí a kôl, tlačidlo na ovládanie herného priebehu. V takom istom štýle sme sa rozhodli po pravej strane umiestniť obchod na nakupovanie veží pre hráča. (1)

Na vyplnenie zvyšného priestoru obrazovky sme s unity Asset Store stiahli vyhovujúce pozadie nastavili textúru skyboxu. (2)

3.4 Menu



Aby sme zachovali náš trojrozmerný nízko polygónový štýl, sme tlačidlá v menu implementovali ako trojrozmerné objekty a samostatné menu prostredie sme umiestnili nad hernú plochu. Na prepínanie pohľadu do menu obrazovky sme pridali na kameru jednoduchú animáciu.

Aby sme u hráča vytvorili pocit responzivity ovládacích prvkov priradili sme skripty aby tlačidlá v menu jemne vystúpili dopredu po tom ako nad nimi prejdeme myšou a aby všetky tlačidlá prehrali zvuk kliknutia

po tom ako sme na nich klikli myšou.

3.5 Hlavný ovládač

Aby sme vedeli centralizovať jednotlivé udalosti

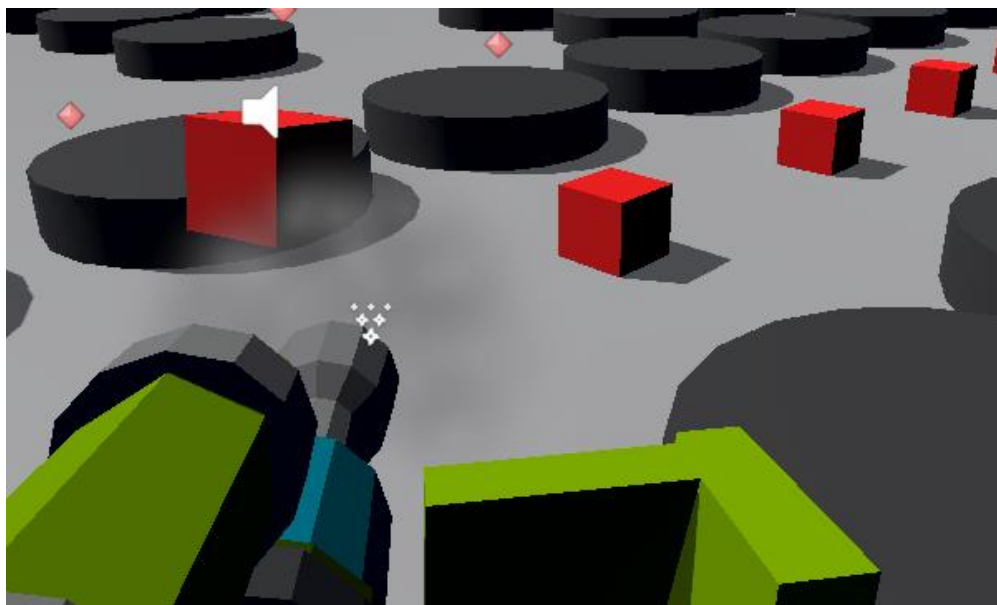
Obrázok 3 Menu

do jedného skriptu sme vytvorili triedu ktorá bude mať za úlohu prepojiť všetky prvky hry, posúvať medzi nimi informácie, sledovať užívateľské vstupy alebo spúšťať určité funkcie.

Keďže sme chceli docieľiť aby bol jeho obsah pre všetky objekty rovnaký nastavili sme väčšinu premenných a funkcií ako verejné a statické.

Vďaka tejto triede vieme z hocijakého iného skriptu v hre upraviť hodnoty alebo zavolať funkciu bez referencie na konkrétny objekt. Napríklad pridať peniaze, pozastaviť hru, odobrať život... Taktiež je zodpovedný za prehrávanie hudby. Pokiaľ je scéna zameraná na menu alebo je pozastavená skript zapne prehrávač hudby s menu hudbou. Pokiaľ je hudba spustená zapne prehrávač hudby s hlavnou hudbou.

3.6 Nepriatelia



Obrázok 4 Nepriatelia

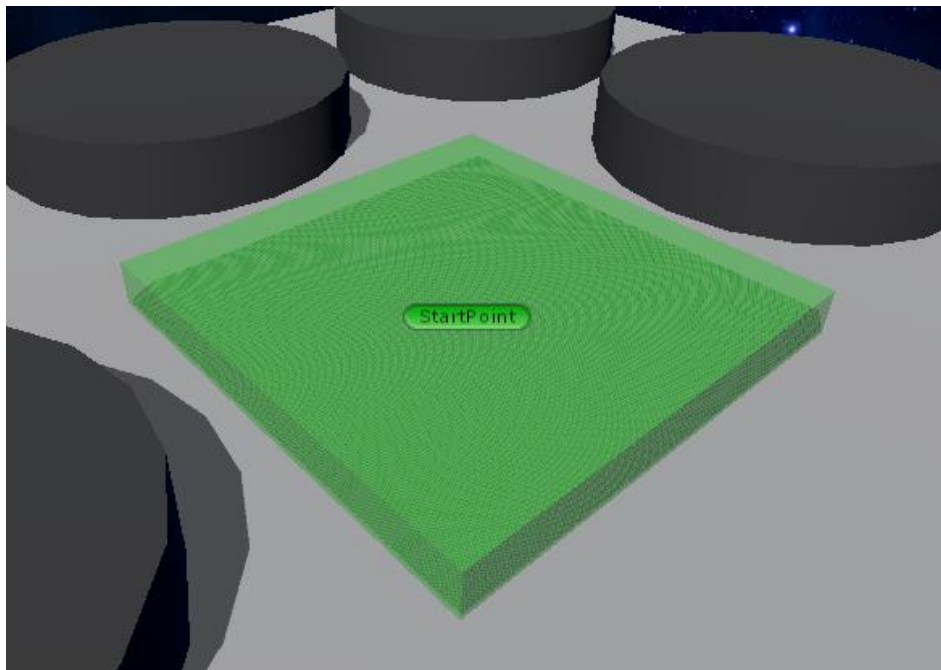
Vzhľad nepriateľov sme sa rozhodli zanechať pomerne jednoduchý. Majú podobu kociek s jednoliatou farbou. Tento dizajn nám umožnil jednoducho upraviť jeho vlastnosti počas priebehu samotnej hry. Na základe úrovne nepriateľa vieme upraviť jeho veľkosť, farbu a rýchlosť.

Ako bolo už vyššie spomínané pohyb nepriateľov je zariadený jednoduchým lineárnym pohybom po prednastavených orientačných bodoch. Pokiaľ nepriateľ dosiahne svoj posledný bod, predpokladáme že dorazil do konca a môže hráčovi ubrať životy. (3)

Pre ďalšie zefektívnenie sa pred začiatkom hry vytvorí verejne dostupné pole orientačných bodov aby si ho nemusel každý nepriateľ hľadať samostatne.

So zámerom urobiť nepriateľov viac živšími sme im pridali zvukové a vizuálne efekty ktoré sa spustia keď nepriateľom odoberieme životy.

3.7 Spawner



Obrázok 5 Začiatočný bod

Pre spawner sme na začiatku zadefinovali konštantný bod kde sa budú vytvárať nový nepriatelia.

S úmyslom ušetrenia procesorových prostriedkov počas priebehu hry sme vytvorili systém ktorý nám pred začatím hry vytvorí konštantný počet neaktívnych nepriateľov a postupne ich aktivuje a deaktivuje podľa potreby.

Pre tento objekt sme vytvorili dodatkovú triedu ktorá v sebe obsahuje informácie o jednotlivých kolách, teda časove rozmedzie medzi jednotlivými nepriateľmi a počty nepriateľov jednotlivých úrovní.

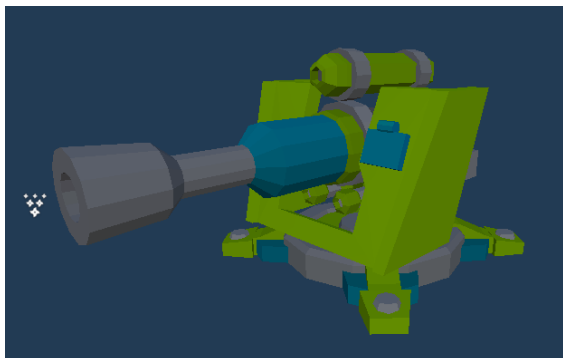
Aby sa nepriatelia aktivovali nezávisle od priebehu ostatných funkcií sme pre túto funkcionality použili špeciálny typ funkcie IEnumerator. Týmto sme dokázali docieľiť aby prebiehal level paralelne od hlavných procesov.

Tomuto objektu sme taktiež priradili úlohu na pozorovanie počtu aktívnych nepriateľov aby sme na základe toho vedeli usúdiť či sme level dokončili. Taktiež počíta koľký level je momentálne aktuálny, respektíve či sme úspešne dokončili všetky levely.

Zadefinovali sme mu funkciu na spustenie ďalšieho levelu ktorá sa dá použiť externe alebo sa zavolá hneď po skončení aktuálneho levelu pokiaľ je zapnutá možnosť automatického spúšťania.

3.8 Veže

Obrázok 6 Kanón



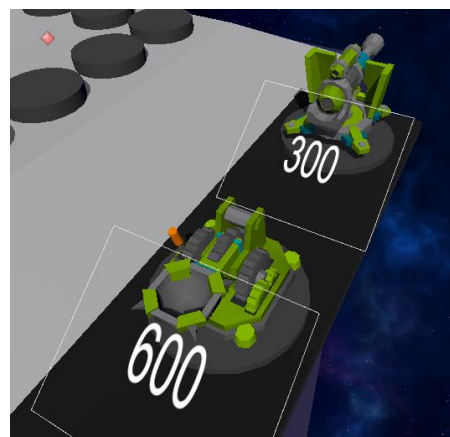
Obrázok 7 Katapult

Keďže veže sú jednou z najpodstatnejších elementov tower defence žánru preto boli aj jednou z najkomplexnejších elementov našej hry na implementáciu.

Keďže nám to zadanie dovolilo, ušetrilo by nám to veľa roboty a zlepši to celkový vzhľad hry rozhodli sme sa že modely a textúry pre veže použijeme s externého zdroja. Tento konkrétny balíček sme si zvolili kvôli tomu že obsahoval všetko čo budeme potrebovať, ponúkal nám na výber hneď niekoľko rôznych typov veží a preto že modely boli ideálne rozdelené na objekty vďaka čomu sme ich vedeli jednoducho zanimovať.

Keďže sme v koncepte do našej hry navrhli niekoľko rôznych veží z ktorých každá svoje vlastné odlišné vlastnosti, výhody a nevýhody, sme sa rozhodli pri ich programovaní použiť dedičnosť. Najskôr sme napísali materskú (šablónovú triedu) v ktorej sme zadefinovali všeobecné parametre ako vzdialenosť dostrelu a všeobecné funkcie ako pravidelné vyhľadávanie nepriateľov v dosahu. Z tejto materskej triedy budú dediť už špecializované skripty pre konkrétne typy veží. Zadefinovali sme v nich ďalej napríklad aký projektíl budú strieľať po nepriateľoch a ďalšie detaily ktoré by sa mohli líšiť od ostatných veží. (4)

Ďalšou podstatnou funkcionalitou veží bol spôsob na ich nakupovanie a umiestňovanie do hracieho poľa. Na implementovanie tejto funkcionality sme na pravej strane od hracej plochy sme vytvorili jednoduchú obchodnú zónu. Pre každý typ veže sme do nej pridali podstavec s modelom príslušnej veže. Po kliknutí na typ veže ktorú chceme zakúpiť sa nám vytvorí daná veža. Všetky jej hlavné skripty sa načítajú ako deaktivované a bude sa riadiť iba podľa skriptu na pokladanie. Po položení sa hráčovi odčítajú peniaze a aktivujú sa všetky hlavné skripty.



Obrázok 8 Obchod veží

3.9 Upgrade systém

Aby sme hráčovi umožnili ďalej rozvíjať rôzne stratégie pridali sme vežiam ďalší skript ktorý hráčovi umožní vylepšovať jednotlivé veže. Kvôli rôznorodosti typov veží sme opäť využili dedičnosť. V materskej triede si definujeme spoločné funkcie, napríklad zobrazenie vylepšovacieho menu po kliknutí myšou na vežu ktorú chceme vylepšiť a premenné ako úroveň vylepšení. Do dcérskych tried sme si zdefinovali už jednotlivé cesty vylepšení a ich ceny. Táto funkcionalita hráčovi taktiež umožní jednotlivé veže predat' a to za zlomok nákupnej hodnoty veže a jej aktuálne nainštalovaných vylepšení.

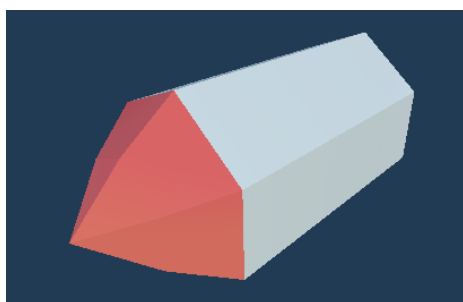


Obrázok 9 Upgrade menu

Menu obsahuje dve tlačidlá každé z ktorých je určené na vylepšovanie jednej z možných ciest vylepšení daného typu veže. Text nad tlačidlom hráča informuje o type vylepšenia a jeho aktuálnej úrovni. Text v tlačidle hráča informuje o cene daného vylepšenia a dôraze na parametre vylepšovanej veže. V pravom dolnom rohu menu obsahuje ešte jedno tlačidlo na predávanie označenej veže. Text v tlačidle na predávanie hráča informuje o cene za ktorú je schopný označenú vežu predat'.

Po kliknutí na konkrétnu vežu ktorú chceme vylepšiť sa aktivuje skript zodpovedný za ovládanie vylepšovacieho systému. Po aktivácii skript hráčovi zobrazí vylepšovacie menu a aktualizuje jeho hodnoty podľa konkrétneho typu veže a jej aktuálne nainštalovaných vylepšení. Pokiaľ je menu aktivované a hráč kurzorom klikne na jedno z tlačidiel tak pomocou eventov je vo vylepšovacom skripte veže zavolá funkcia ktorá zistí či má hráč dostatok peňazí na dané vylepšenie a prípadne vylepšenie vykoná a odpočíta hráčovi peniaze. V prípade že hráč klikne na tlačidlo na predávanie, vežu vymaže a pripočíta hráčovi zodpovedajúce množstvo peňazí.

3.10 Projektily



Obrázok 11 Náboj



Obrázok 10 Bomba

Kvôli rôznorodosti projektílov ktoré môžu naše veže po nepriateľoch strieľať sme v skriptoch využili dedičnosť. Každý projektíl obsahuje premennú na silu útoku a svoju rýchlosť ale trajektórie letu projektílu a štýly útoku sa líšia. Napríklad náboj z kanónu sa pohybuje priamo a poškodzuje iba toho nepriateľa ktorého trafil a bomba z kanónu sa pohybuje po oblúkovitej trajektórii a poškodzuje všetkých nepriateľov v určitom dosahu.

3.11 Užívateľské rozhranie



Obrázok 12 Ovládací panel

Ďalšou z podstatných častí hry čo sme pridali do hry bolo užívateľské rozhranie. Bolo potrebné z toho dôvodu aby užívateľ vedel ovládať chod hry a vyčítať z neho dôležité informácie ako počet peňazí a života. Modely sme opäť vytvorili z jednoduchých objektov aby sme zachovali náš trojrozmerný nízko polygónový štýl. Všetky prvky užívateľského rozhrania sú prepojené a ovládame ich pomocou Controller skriptu.

Prvým podstatným elementom ktorý sme museli implementovať bolo tlačidlo na ovládanie rýchlosti času. Vizuálne aj funkčne má štyri štádiá - pozastavené, normálna rýchlosť, dvojnásobná rýchlosť a štvornásobná rýchlosť. Rýchlosti čau vieme vieme ovládať buď kliknutím myšou na tlačidlo alebo stlačením klávesy escape ktorá nás presunie do hlavného menu. Tlačidlo sa taktiež automaticky prepne do pozastaveného štádia pokiaľ úspešne dokončíme level a nemáme zapnutú funkciu automatického zapínania levelov.

Tlačidlo na automatické zapínanie levelov sme implementovali aby sme pre hráča spohodlnili priebeh hry. Do tlačidla sme implementovali systém prepínača, respektíve po kliknutí tlačidlo prejde do zapnutého štádia a po ďalšom kliknutí prejde naspäť do vypnutého štádia. Podľa svojho aktuálneho štádia dáva ostatným skriptom vedieť či sa má automaticky začať ďalší level.

Do užívateľského rozhrania sme implementovali aj niekoľko dôležitých indikátorov o stave hry. Jeden z najpodstatnejších z nich je indikátor života. Vyrobili sme ho jednoducho z červeného kvádra. Počet aktuálnych životov vieme znázorniť tak že jednoducho zmenšíme alebo zväčšíme tento kváder.

Ďalšie indikátory ktoré sme pridali boli počítadlo peňazí a ukazovateľ aktuálneho levelu. Dizajn mal predstavovať štýl sedem segmentového displeja. V našom prípade každá číslica obsahuje sedem segmentov ktoré vieme osobitne aktivovať a deaktivovať pomocou skriptu. Ďalej sme podľa potreby skombinovali niekoľko týchto číslic do jedného objektu a priradili sme mu skript ktorý bude upravovať jednotlivé číslice podľa čísla ktoré mu zadáme ako vstupný parameter. Aktualizácia číselných displejov bude ovládaná pomocou Controller skriptu podľa potreby.

4 VÝSLEDKY PRÁCE A DISKUSIA

V tejto časti práce sme sa venovali opisu vývoja 3D hry v Unity a popísali sme vlastné zistenia, výsledky a pozorovania získané počas jej vývoja. Najprv sme sa venovali návrhu herného sveta a jeho vizualizácii v Unity pomocou rôznych techník a nástrojov, ako sú napríklad ProBuilder, alebo Asset Store. Potom sme sa venovali programovaniu herných mechanizmov, ako sú ovládanie hráča, kolízie, ovládače inteligencie nepriateľov a podobne. Pri programovaní sme používali jazyk C# a vývojové prostredie Microsoft Visual Studio 2022.

Počas vývoja hry sme sa stretli s rôznymi problémami a výzvami, ktoré sme museli riešiť. Napríklad vytváranie efektov a animácií, optimalizácia výkonu, alebo zabezpečenie užívateľskej interakcie a použiteľnosti hry. Všetky tieto výzvy sme museli riešiť s ohľadom na to, aby hra bola zaujímavá a zábavná pre hráčov. Vzhľadom na rôzne neočakávané problémy s ktorými sme sa počas vývoja stretli bolo potrebné sa naučiť a zdokonaľiť sa v mnohých oblastiach softvérového inžinierstva a herného vývoja. Napríklad koncepty event systémov, polymorfyzmus v Unity alebo paralelné funkcie. Napriek tomu že s prvotného hľadiska to boli prekážky sa po dôkladnom preskúmaní problémov podarilo nielen z hľadiska celkovej komplexity posunúť hru na vyššiu úroveň ale aj zdokonaľiť naše programátorské zručnosti a vedomosti.

V priebehu tvorby hry sme ju priebežne testovali a zhodnocovali jej kvalitu a použiteľnosť. O výsledkoch sme priebežne diskutovali s konzultantom a porovnali sme ich s výsledkami iných autorov a hier. Zistili sme, že vývoj 3D hier v Unity je zložitý a časovo náročný proces, ale pomocou rôznych nástrojov a techník si každý záujemca o tvorbu počítačových hier vie vytvoriť svoju vlastnú kvalitnú a zábavnú hru. Aj keď sa proces tvorby hry na začiatku mohol zdať veľmi náročný a pokročilý na konci sme zistili že s minimálnymi začiatočnými skúsenosťami je v celku jednoduché sa do toho dostať a konečný výsledok bol veľmi uspokojivý a skúsenosti a vedomosti ktoré sme tvorbe nabrali boli maximálne hodnotné.

5 ZÁVER PRÁCE

Cieľom tejto technickej dokumentácie bolo popísať postup vývoja 3D tower defence hry v Unity, vysvetliť použité technológie a nástroje a zhodnotiť dosiahnuté výsledky. Na základe analýzy sme stanovili hlavný cieľ, ktorým bolo vytvorenie hrateľnej verzie 3D tower defence hry v Unity. Pre splnenie tohto cieľa sme si stanovili čiastkové ciele, ako napríklad tvorba herného prostredia, implementácia základných herných mechanizmov, ako aj pridanie vizuálnych a zvukových efektov.

V rámci práce sme použili objektovo orientovaný programovací jazyk C# a nástroje Unity ProBuilder a Asset Store. Použitie týchto nástrojov nám výrazne zjednodušilo a zrýchlilo tvorbu herného prostredia a umožnilo nám ľahšiu prácu s 3D objektmi. Využitie jazyka C# nám umožnilo napísať jednoduchý a rýchly nízkoúrovňový program s výbornou čitateľnosťou.

Po implementácii všetkých cieľov a herných prvkov sme úspešne vytvorili hrateľnú verziu 3D tower defence hry v Unity s funkčnými základnými hernými mechanizmami. Výsledky nám potvrdili správnosť zvolenej metodiky a použitých technológií. Získané poznatky môžu byť použité na ďalšiu tvorbu hier a aplikácií s 3D prostredím a objektovou štruktúrou.

V budúcnosti by sme mohli využiť získané poznatky na implementáciu ďalších funkcií do hry, ako napríklad rôzne varianty nepriateľov alebo vylepšiť vizuálne efekty a zvukové prvky. Výsledky nám ukázali, že použitie objektovo orientovaného programovacieho jazyka a Unity Asset store nám umožňuje rýchlejší a efektívnejší vývoj hier a aplikácií. V závere by sme chceli zdôrazniť, že práca na tejto hre nám poskytla veľa cenných skúseností a poznatkov, ktoré môžeme v budúcnosti využiť pri našich ďalších projektoch. Využitie nášho kódu by tiež mohlo byť prínosom pre ostatných vývojárov, ktorí by mohli použiť niektoré z našich riešení alebo nápadov.

Celkovo môžeme povedať, že vývoj 3D tower defence hry v Unity bol úspešný a prínosný nielen pre nás, ale aj pre širšiu vývojársku komunitu.

ZHRNUTIE

Táto práca sa zaoberala vytvorením 3D hry v Unity engine s využitím nástrojov ako Unity ProBuilder a Microsoft Visual Studio. Cieľom bolo vytvoriť plne funkčnú a esteticky príjemnú hru s pomocou týchto nástrojov. Metodika práce zahŕňala tvorbu herného prostredia, herných postáv, mechaniky hry a programovania herného prostredia. Získané výsledky potvrdili, že využitie týchto nástrojov v kombinácii s programovaním môže viesť k vytvoreniu kvalitnej 3D hry. V budúcnosti by bolo možné rozšíriť hernú mechaniku a pridať ďalšie funkcie, ako napríklad multiplayer režim alebo ďalšie herné úrovne. Zhrnutím možno konštatovať, že táto práca úspešne dosiahla svoj cieľ a predstavila možnosti využitia nástrojov vytvorených pre tvorbu 3D hier. Vytvorená 3D tower defence hra je plne funkčná a poskytuje hráčom príjemné vizuálne a herné zážitky. Získané výsledky potvrdili, že využitie týchto nástrojov v kombinácii s programovaním môže viesť k vytvoreniu kvalitnej 3D hry. Hra bola testovaná a overená, že funguje správne. Hráči môžu v hre plánovať obranné stratégie a bojovať s rôznymi druhmi nepriateľov v pútavej 3D grafike.

ZOZNAM POUŽITEJ LITERATÚRY

1. **Brackeys.** YouTube. *Brackeys - How to make a Tower Defense Game.* [Online] 2017.
<https://www.youtube.com/playlist?list=PLPV2KyIb3jR4u5jX8za5iU1cqnQPmbzG0>.
2. **FattyWar.** Unity Asset Store. *Fatty Poly Turret Free.* [Online] 14. 10 2019.
<https://assetstore.unity.com/packages/3d/props/weapons/fatty-poly-turret-free-155251>.
3. **Avionx.** Unity Asset Store. *Skybox Series Free.* [Online] 4. 1 2022.
<https://assetstore.unity.com/packages/2d/textures-materials/sky/skybox-series-free-103633>.
4. **Brackeys.** YouTube. *Unity NavMesh Tutorial - Basics.* [Online] 2019.
<https://youtu.be/CHV1ymlw-P8>.

ZOZNAM PRÍLOH

Príloha A Používateľská príručka	26
Príloha B Kanón.....	30
Príloha C Katapult.....	31
Príloha D Prehľad prostredia	32
Príloha E Sedem segmentový modul	33
Príloha F spustiteľný exe súbor.....	34
Príloha G Video ukážka hry	35

PRÍLOHY

Príloha A Používateľská príručka

1. Ovládanie

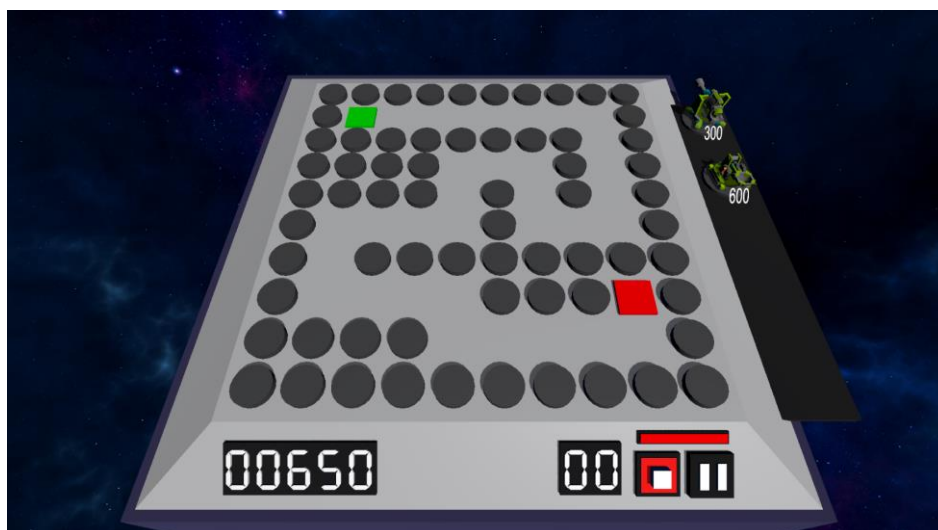
- Väčšina mechaník sa ovláda pomocou myši
- ESC – návrat do menu / prerušenie akcie
- MEDZERA – pozastavenie / spustenie hry

2. Hlavné menu



- Restart – reštartovanie hry od začiatku
- Play – začiatok hry / návrat do hry
- Quit – ukončenie hry

3. Hlavné prostredie



- Tlačidlo na ovládanie toku hry



Pomocou tohto tlačidla vieme nastaviť rýchlosť hry. Ovládal sa myšou. Pozastavenie vieme ovládať medzerníkom. Môžeme ho prepínať medzi štyrmi štádiami pauza/1x/2x/4x.

- AutoPlay tlačidlo



Keď ho aktivujeme levely sa budú spúšťať automaticky

- Počítadlo levelov



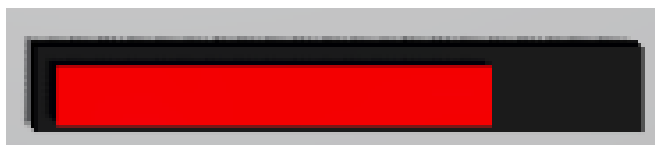
Ukazuje aktuálny level

- Počítadlo peňazí



Ukazuje koľko má hráč k dispozícii peňazí

- Indikátor životov



Ukazuje koľko hráčovi ostáva života

- Začiatkové a koncové body nepriateľov



Znázorňujú akým smerom sa budú nepriateľa pohybovať, kde sa budú objavovať a kde budú hráčovi uberať životy.

- Obchod na veže



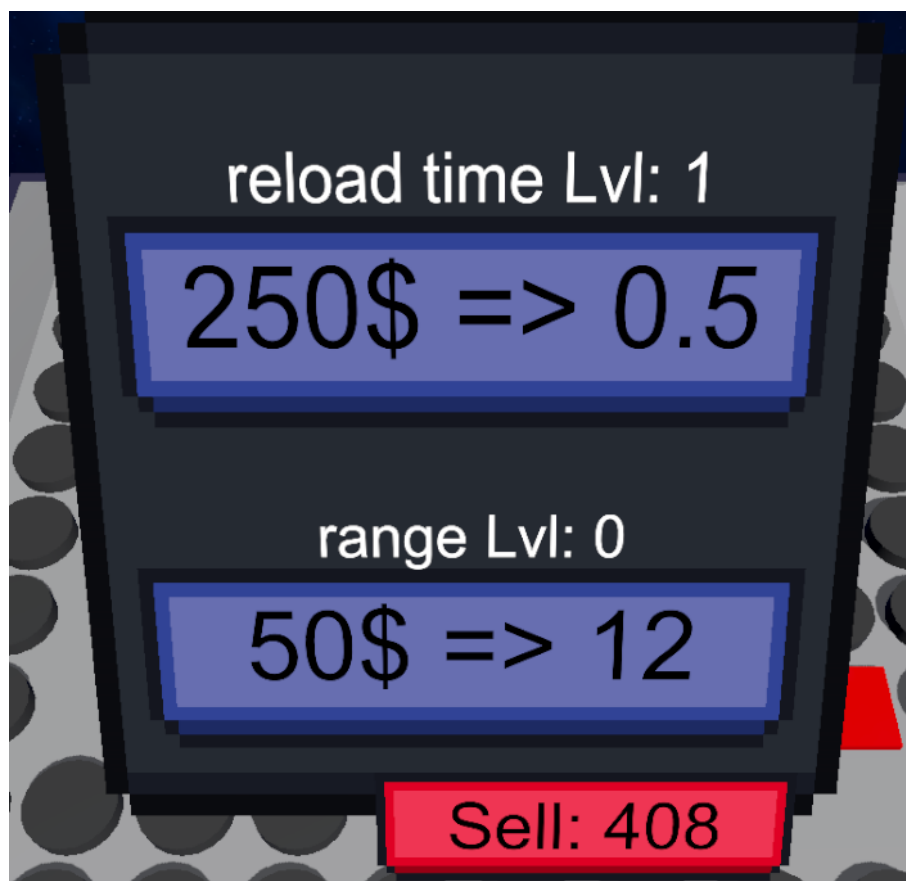
Ukazuje hráčovi ktoré veže si môže zakúpiť a za akú cenu. Po kliknutí myšou na požadovanú vežu sa hráčovi spustí režim úprav kde môže vybranú vežu postaviť za predpokladu že má dosť peňazí.

- Obranné body



Na každý s týchto platforiem môže hráč postaviť práve jednu vežu.

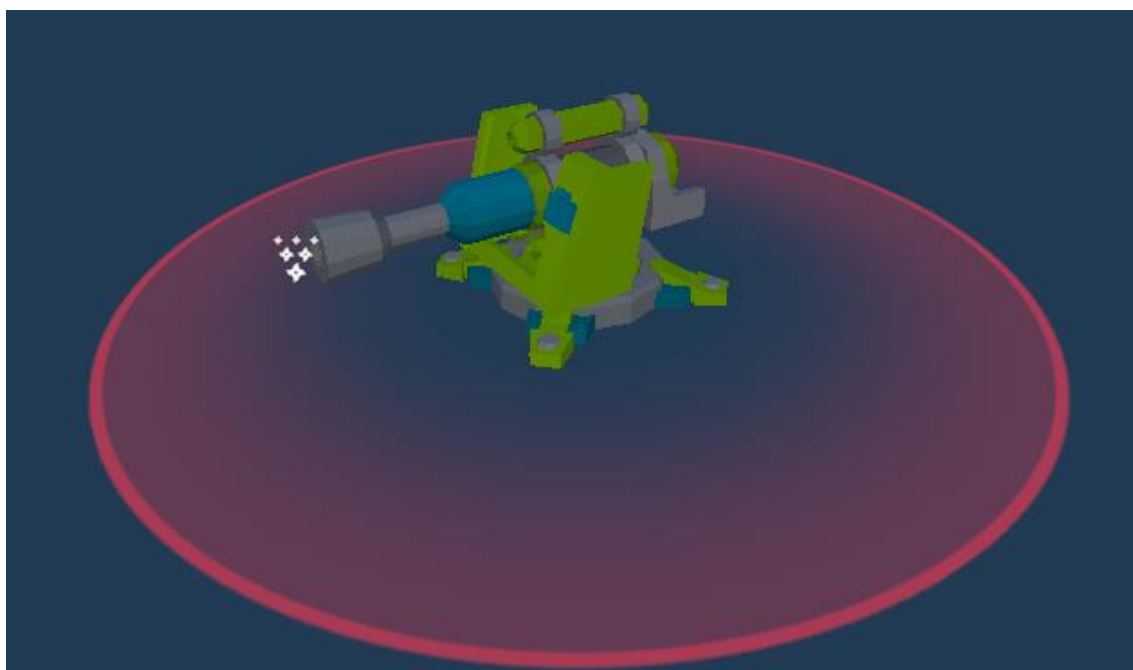
- Vylepšovacie menu

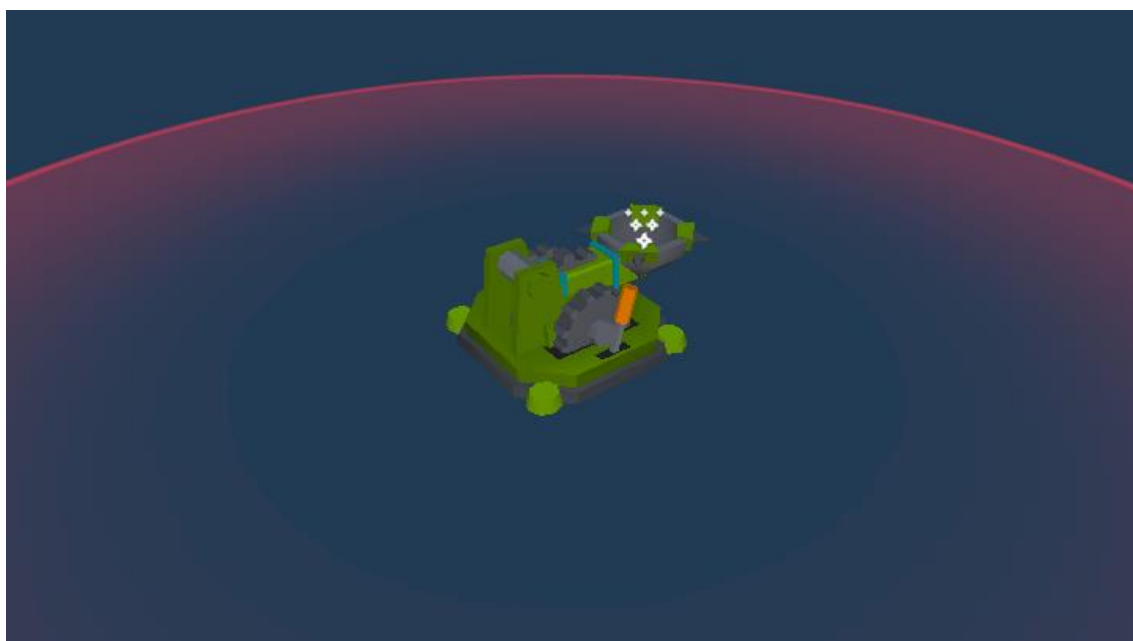


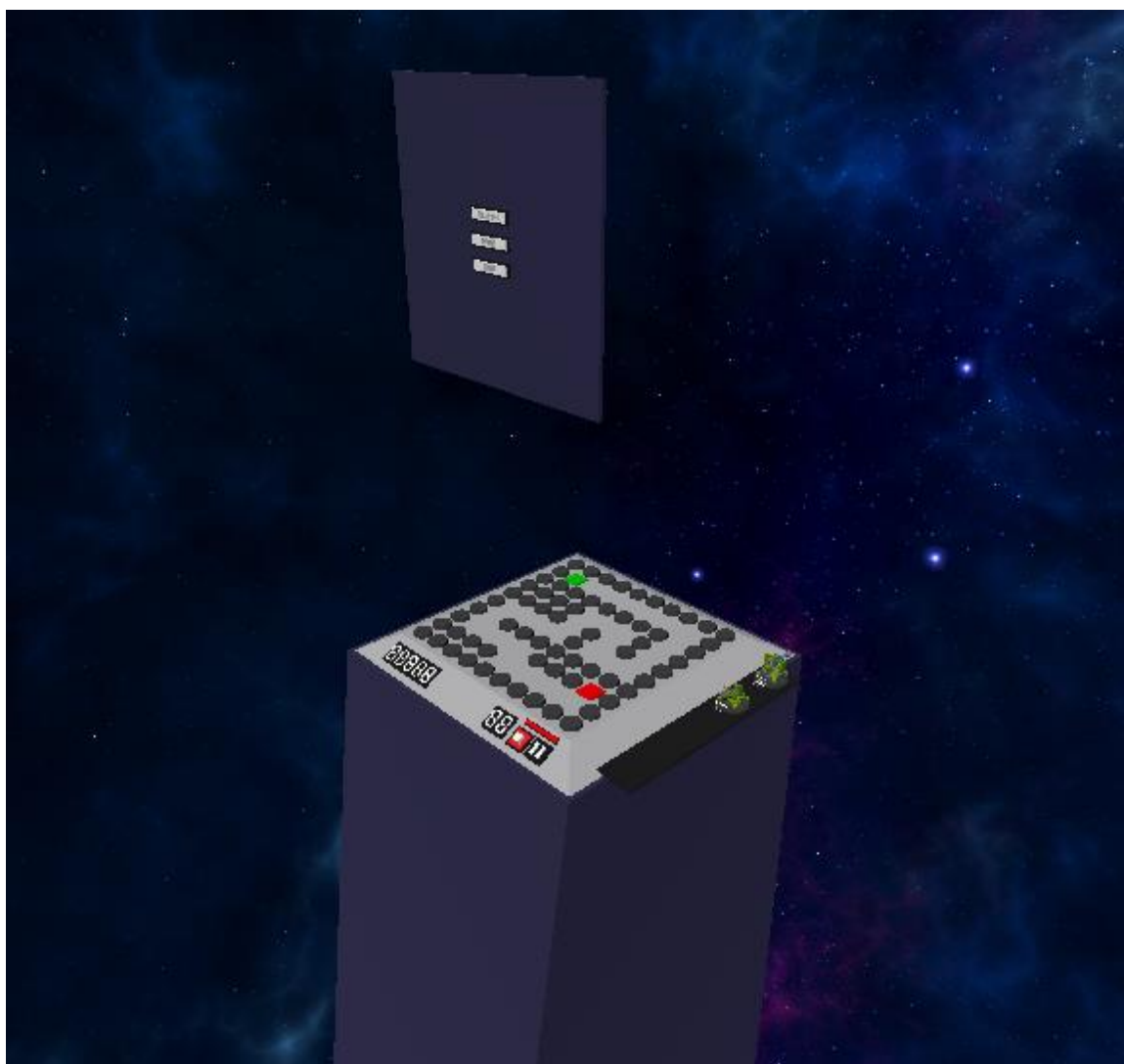
Po kliknutí na už zakúpenú vežu sa zobrazí vylepšovacie menu. Poskytuje mu informácie o úrovni vylepšení veže, dostupných vylepšeniach, cenách vylepšení a cene za ktorú môže hráč vežu predat’.

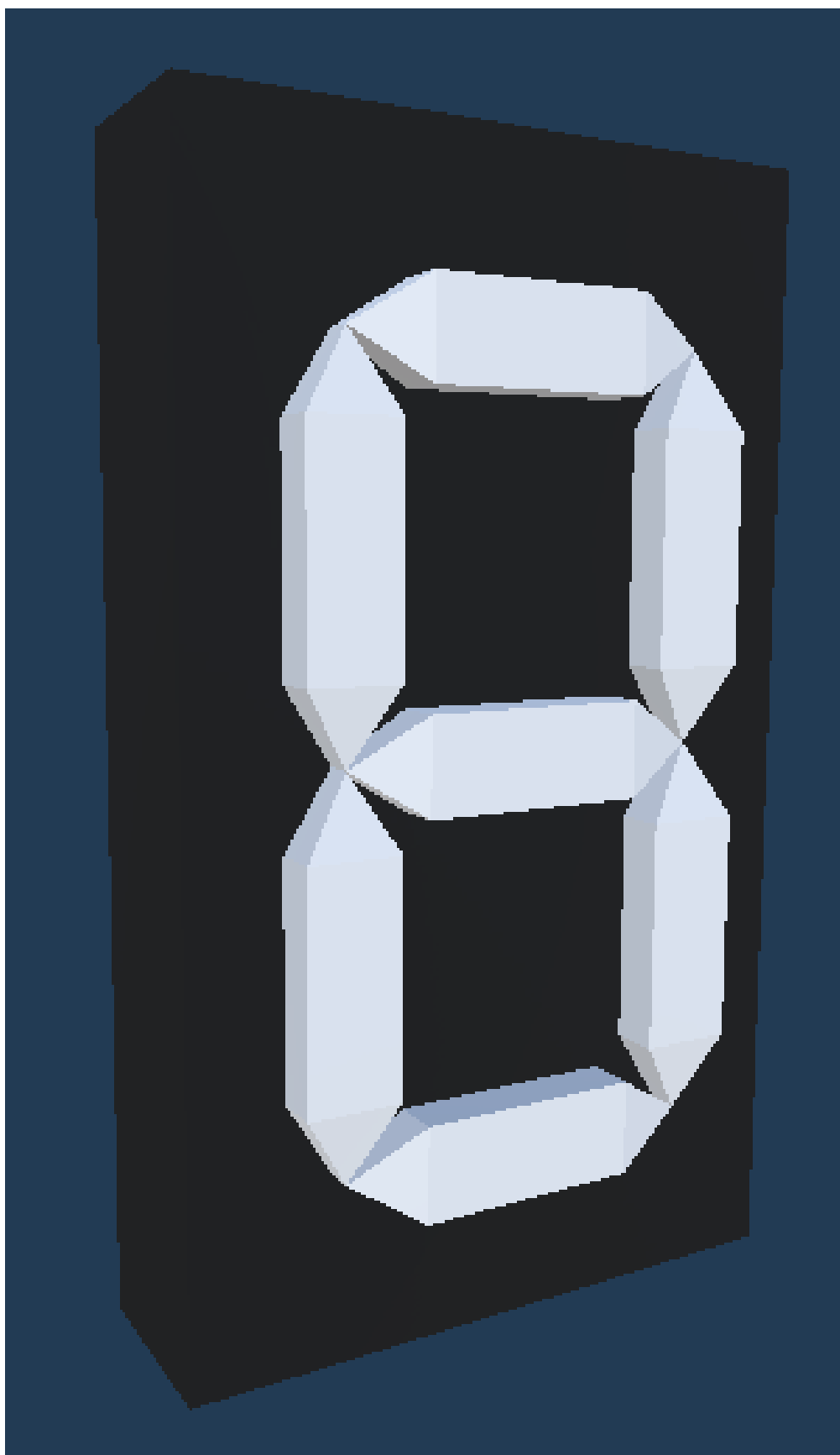
4. Priebeh hry

- Hra sa spustí po kliknutí na tlačidlo Play v hlavnom menu
- V hlavnom prostredí si hráč postaví veže podľa svojich potrieb.
- Nepriateľské vlny sa spustí pomocou tlačidla na ovládanie toku hry.
- Postupovaním v hre a obraňovaním proti nepriateľom si hráč zarobí peniaze ktoré spätne investuje do svojich obrán.
- Cieľom hry je prejsť všetky levely bez toho aby hráč prišiel o všetky svoje životy.









- Vid' disk

- Vid' disk