

Szakképesítés neve: Szoftverfejlesztő

OKJ száma: 54 213 05

SZAKDOLGOZAT

Recaller c. játék

Rostás Péter
témavezető

Füzi Bálint Krisztián
2/14.E

Tartalomjegyzék

A felhasználói dokumentáció.....	2
A program általános specifikációja	2
Rendszerkövetelmények	2
Hardver követelmények.....	2
Szoftver követelmények	2
A program telepítése.....	3
A program használatának a részletes leírása	5
Grafikai Elemek és Hangok.....	5
Játékmenet	8
Fejlesztői dokumentáció	11
Témaválasztás indoklása.....	11
Alkalmazott fejlesztői eszközök.....	11
Használt programok	11
Adatmodell leírása	12
Programszerkezet	12
Adatmodellek.....	14
Tesztelési dokumentáció	22
Továbbfejlesztési lehetőségek.....	23
Történet üzemmód.....	23
Grafikai fejlesztések.....	23
Változatos ellenségek	23
Játékszabályok kiegyensúlyozása	23
Irodalomjegyzék, forrásmegjelölés.....	24

A felhasználói dokumentáció

A program általános specifikációja

Nem kell sokáig keresnünk, hogy találjunk olyan filmet vagy videójátékot ami a zombi-apokalipszis témáját dolgozza fel. Ezek a játékok és filmek azt akarják szimulálni, hogy miképp fordulna fel az emberek élete, amennyiben élőhalottak lepnék el városukat, bolygójukat, akiknek célja, hogy megöljék a még élő lakosokat. Az ezzel foglalkozó játékok általában az olyan emberek bőrébe helyezik a játékost (a felhasználót), aki a túlélők közé tartozik. A „Recaller” c. játékkal is ezt a témát szerettem volna feldolgozni.

A játék célja minél tovább túlélni a ránk támadó zombik ellen. Ezek a zombik ú.n. hullámokban jönnek: a játékban van nyugalmi időszak, illetve támadási időszak, utóbbit hívom hullámnak, hiszen ilyenkor támadnak a zombik. A túléléshez rendelkezésünkre álló eszközök:

- Lőfegyverek
 - Többféle fegyver vehető, mindegyik más erősséggel. A fegyverekhez lőszert kell venni.
- Barikádok
 - Barikádok építhetők a nyugalmi időszakban, ha elpusztítják a zombik az adott védelmet, akkor újra kell vásárolni.
- Képességek (Sebesség, Életpont, Stb.)
 - A játékos a nyugalmi időszakban fejlesztheti a képességeit, amivel gyorsabban mozoghat, több támadást élhet túl vagy gyorsabban tüzelhet a fegyverével.

A játék végét az jelenti, ha a játékost megölik a zombik, ekkor a teljesítményünk alapján egy pontszámot kapunk és kezdhethjük is újra a játékot.

Rendszerkövetelmények

Hardver követelmények

Minimum követelmények:

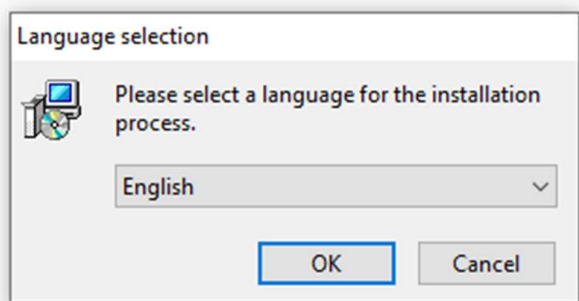
- CPU: Intel Core 2 Duo E6550 2.33GHz
- GPU: Intel HD3000 / Nvidia GeForce 8600GT (256MB) / AMD/ATI Radeon HD3650 (256MB)
- RAM: 2GB
- Tárhely: 200MB

Szoftver követelmények

- Operációs rendszerek:
 - Windows 7/8/10 (32 vagy 64 bit)
- .NET Framework 4.8
- DirectX 9

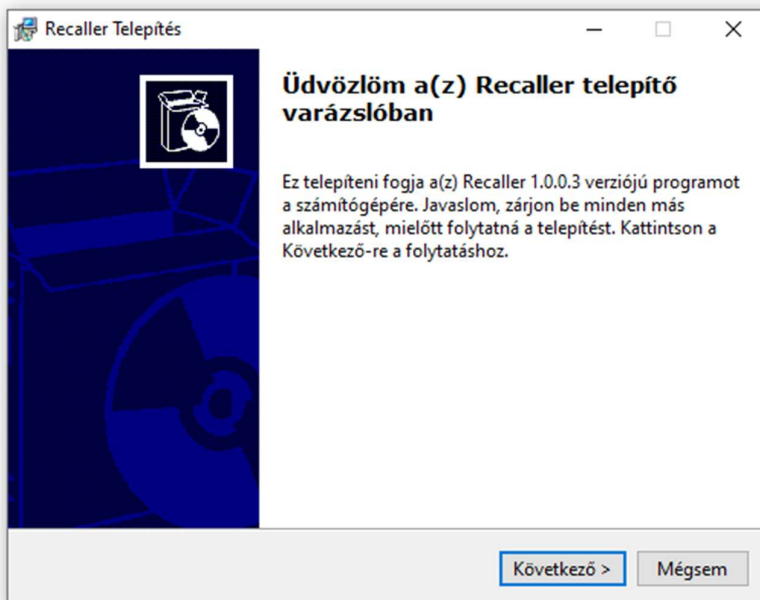
A program telepítése

1. A program telepítése a mellékelt CD-n található setup.exe megnyitásával kezdődik.
2. A fájl megnyitása után egy nyelvválasztó fogad minket, itt a számunkra szükséges nyelvet választhatjuk ki. A választható lehetőségek: angol/magyar.



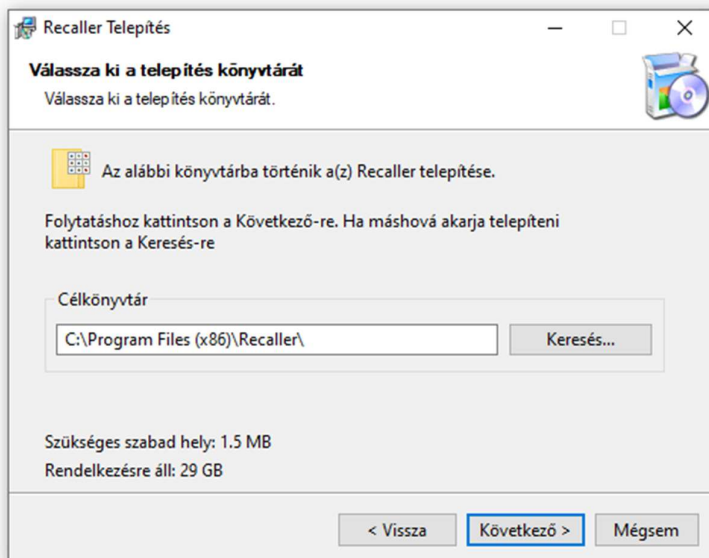
ábra 1 nyelv választás

3. Ezután a megnyitott telepítő fogad minket az általunk kiválasztott nyelven, nyomjunk a *következő* gombra.



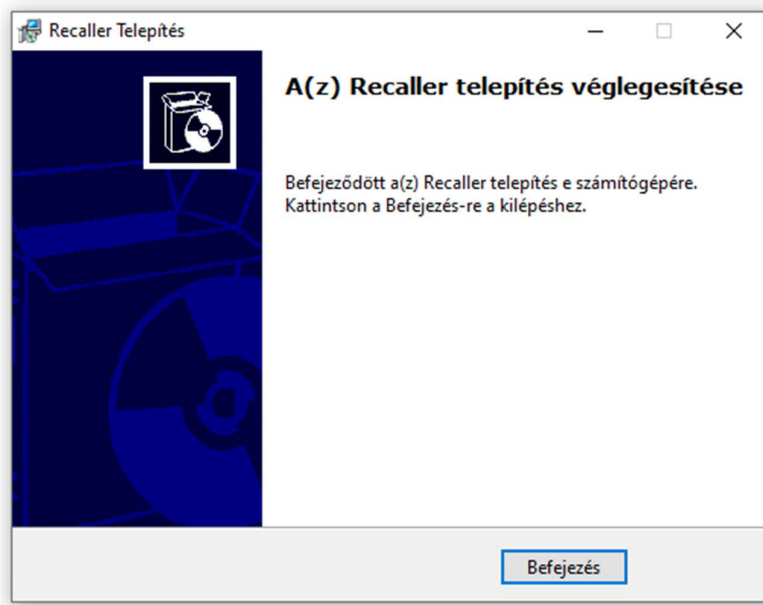
ábra 2

4. A telepítő felkínálja nekünk a lehetőséget, hogy hova szeretnénk telepíteni a programunkat, választhatunk más helyet, de az alapértelmezett helyre is telepíthetjük, a *következő* gomb megnyomásával.



ábra 3

5. Lehetőségünk van kiválasztani, hogy a program létrehozzon-e asztali és/vagy Start menü ikonokat. Miután kiválasztottuk a számunkra megfelelő dolgokat, nyomjunk rá a *következő* gombra.
 - a. Ha kiválasztottuk a Start menü könyvtár lehetőséget, akkor van lehetőségünk a következő lépésben elnevezni a könyvtárat. Itt is nyomjunk a *következő* gombra, ha minden megfelel.
6. A lépések után egy rövid telepítő folyamat fog lefutni. Várjuk meg.
7. A folyamat lefutása után egy megerősítő képernyő fogad minket, nyomjunk a *befejezés* gombra és már indíthatjuk is a programot.



ábra 4

A program használatának a részletes leírása

Grafikai Elemek és Hangok

Főmenü:

A játékot elindítva a játék fő menüje tárul elénk. Ebből a menüből lehet irányítani a programot. A szokványos kurzor helyett egy célkereszt jelenik meg, ezzel tudjuk kiválasztani a számunkra szükséges menüpontot. A kurzort a menü gombjai fölé mozgatva megváltozik a kijelölt gomb kinézete. Ezzel is jelezve a felhasználónak, hogy mire fog rákattintani.

Bármelyik menüpontra kattintva egy újabb menü tárul elénk egy vékony fehér sávval mögötte. A kattintás után a kiválasztott gomb kijelölve marad és a többi gomb elsötétül. Ezekben a menükben a korábbi gombok alpontjait találhatjuk. Az almenün kívülre kattintva a játék kijelzője visszarendeződik eredeti állásába.

A gombok és almenüik leírása:

- New game (új játék)
 - Ebben a menüben új játék kezdhető.
- Continue (folytatás)
 - Korábbi játékok folytathatók
- Settings (beállítások)
 - Music/Effect gomboknál a hangereje a zenének/effekteknak beállítható egy csúszkával, ami fölött a hangerőt számosítva is láthatjuk, az egyszerűség érdekében. A hangerő 0-ra állításával lenémíthatjuk az adott hangtípust.
 - Törölni lehet a játék mentéseit.
 - Meg lehet nézni a korábbi pontszámokat.
- About (a játékról)
 - Ebben a menüpontban látható a játék készítője, a készítés éve és az intézmény neve ahol a játék készült.
- Quit (kilépés)
 - A játékból itt lehet kilépni egy megerősítést követően

A háttérben különböző grafikák váltogatják egymást, hogy változatossá tegyék a kijelzőt. A háttérben zene is játszódik, illetve kattintáskor lövés hangot ad a játék a kijelölés megerősítéseként.

Játéklablak:

A játék elindításakor az adott módtól és pályától függően más-más ablak tárulhat elénk, de vannak olyan elemek, amik konstansak minden játék ablakban. Ezekre az elemekre felváltva, folyamatosan érdemes figyelni, hogy pontosan tudja a felhasználó, a játék állapotát. A hullám közben (amikor támadnak a zombik) és a nyugalmi időben is másképp néz ki az ablak elrendezése és az elemei.

A hullám közben a képernyőn olyan elemek vannak, amik a játékmenet részei. A középpontból kifelé haladva érdemes figyelni a játékot és a játékelemeket. A játék közepén ugyanis a játékos karakter látható a hozzá közel lévő, rá vonatkozó adatokkal, ezek az adatok:

- Életcsík
 - Ezt egy vékony hosszú téglalapként mutatja a játék a játékos karakter fölött, a színe és a telítettsége jelzi, hogy a játékosnak mennyi életpontja van hátra, milyen messze van a haláltól (a játék elvesztésétől):

- Zöld - Nem sérült / alig sérült.
 - Citromsárga - Enyhén sérült, legalább az életpontok fele hátra van.
 - Narancssárga - Nagyon sérült, az életének több, mint felét elvesztette.
 - Piros - Kritikusan sérült, közel áll a halálhoz, az életének $\frac{3}{4}$ -ét már elvesztette.
- Jelenlegi fegyver neve
 - A játékos alatt található, a lőszerrel együtt. A játékban a különböző fegyvereknek nevei vannak és tulajdonságai. Mindegyiknek más-más tulajdonsága van, amik próbálnak közel állni a valós verziójukhoz, így a név eléggé beszédes.
 - Jelenlegi fegyverben található lőszer mennyisége
 - A lőszer el tud fogyni, kivéve, ha végtelen szimbólumot látunk, ebben az esetben végtelen lőszerünk van.



ábra 5 játékos tulajdonságai

A játékostól távolabb pillantva láthatjuk a különböző elemeket a pályán, amik merőben elütnek a játék háttérétől a talajtól. A talajon található minden más elem, aminek a széléhez érve sötétség fogad minket, innen jönnek a zombi ellenségek. A többi itt található elem:

- Tereptárgyak
 - Olyan elemek, amik nem mozognak a pályán és valami valós tárgyra hasonlítanak. Például: doboz, szögesdrót, téglafal, cölöpkerítés
- Karakterek
 - Olyan elemek, amik mozognak a pályán, általában a játékost közelítve. Ezek általában különböző fajta zombik.
- Kilőtt golyók
 - Apró szürke pontok vagy nagyobb lövedékek a kijelzőn, amik a kilövés irányába mozognak, és nem tudják megsebezni a játékost.



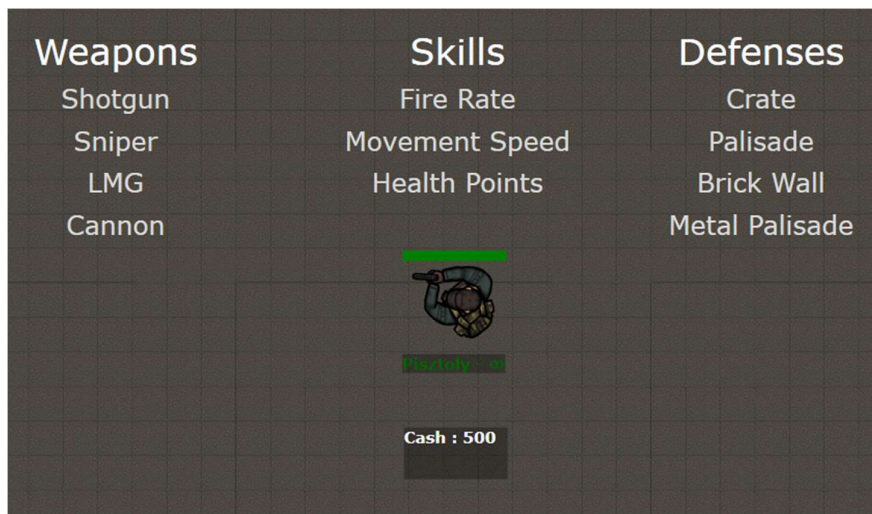
ábra 6 golyók, tereptárgyak, zombi

A korábban említett nyugalmi üzemmódban a megszokottól más kijelzőt láthatunk.

Itt az alábbi dolgokat tudjuk felfedezni a kijelzön:

- Weapons ablak
 - Az angol fegyverek szóból itt lehet fegyvereket vásárolni a játék nyugalmi időszakában, továbbá lőszereket is itt lehet vásárolni az adott fegyver nevére kattintva
 - Shotgun – sörétes puska
 - Sniper – távcsöves puska
 - LMG – géppuska
 - Cannon - ágyú
- Skills ablak
 - A képességeinket fejleszthetjük itt a képességekre kattintva
 - Fire Rate – tüzelési sebesség
 - Movement speed – mozgási sebesség
 - Health Point – életpontok

- Defenses ablak
 - Crate – fa doboz, át lehet fölötte löni
- Cash ablak
 - Itt láthatjuk mennyi pénzünk van.



ábra 7 vásárló menü

Játékmenet

A játék kezdetén teljes életponttal, egy pisztollyal - amiben végtelen lőszer van – és zombikkal kezdünk, ugyanis egy hullám elején találjuk magunkat. Ezt az első hullámot (és az utána következőket) csak akkor tudja megnyerni a felhasználó, ha elsajátítja a játék irányításait, amik:

- Mozgás:
 - W – Fel
 - A – Balra
 - S – Le
 - D – Jobbra
 - Ezek együttes lenyomásával átlósan is lehet mozogni.
- Fegyverváltás (hullám közben):
 - Q és E gombokkal lehet váltogatni a fegyverek között, mintha egy listában nézegetnénk az elemeket
 - Q – Visszafelé váltás
 - E – Előre váltás
- Lövés (hullám közben):
 - Bal egérgomb kattintással és/vagy folyamatos lenyomásával is lehet löni.
- Építés (két hullám közötti felkészülési időben):
 - Szóköz gombbal amerre az egér mutat, le lehet rakni a menüben kiválasztott tereptárgyat.
- Vásárló menü navigálása (két hullám között):
 - Az bal egérgomb kattintásával és az adott elemre mutatással lehet a kiírt dolgot megtenni, például: vásárlás, karakterfejlesztés

- Kilépés:

- ESC

Az irányítások mellett figyelnie kell a felhasználónak a játékot nehezítő dolgokra és a játék tényezőire.

Ezek:

- Zombik:

- A zombik folyamatos veszélyt jelentenek a játékos karakterre. A zombik fő célja megsebezni és megölni a játékos minden áron. A játékos követik a pályán folyamatosan és a játékos ellentámadásaira nem reagálnak. Ez utóbbi annyit tesz, hogy a lövések ugyan megölik őket, de rájuk tüzelve a játékos továbbra is követik. Továbbá a köztük és a játékos közé kerülő akadályokat megpróbálják elpusztítani és/vagy kikerülni. Megölésük rájuk lövéssel lehetséges és a következő nyugalmi üzemmód az adott hullám összes zombijának megölésével érhető el

- Életpont:

- A játékosok, zombik, tereptárgyak is rendelkeznek életponttal. A tereptárgyaknak vagy az ellenségeknek, ha az életpontja lecsökken teljesen, akkor eltűnnek a pályáról. A játékos életpontja viszont sokkal fontosabb. A játékos a zombik tudják megsebezni azzal, hogy közel érnek hozzá és megütik. A játékosnak legfontosabb teendői között kell lennie annak, hogy ez ne történjen meg, hiszen ha sokat veszít az életéből a játékos, akkor elveszíti a játékot. A játékos életének jelenlegi állapotát a korábbi ábra alapján tudja számon tartani a felhasználó. Az adott hullám végével ez a lehetséges maximumra töltődik vissza

- Fegyverek és lőszer:

- A fegyvereket a zombik megölésére lehet használni, de a játékosnak jól számon kell tartania a birtokában lévő fegyvereket. A kezdő fegyver egy pisztoly végtelen lőszerrel, alacsony sebesséssel, közepes lövési sebességgel. Ha a játékos nem vesz új fegyvereket a zombik hamar le tudják rohanni a későbbiekben. Továbbá ha a játékos nem vesz lőszert a fegyverekbe, akkor nem tudja használni őket. Emellett minden fegyvernek meg vannak a maga tulajdonságai, a sörétes puska egyszerre több kis lőszert lő összevissza, a mesterlövész pontosan, de lassan lő erős lövéseket, a gépfegyver sokat lő gyorsan, de nagyon összevissza. Az erősebb fegyverek és lőszereik drágábbak.

- Meglévő tereptárgyak, építhető tereptárgyak:

- A tereptárgyak akadályozzák a játékos és a zombikat is. Mindkettő nekiütkezve nem tud tovább mozogni és ki kell kerülnie vagy – a zombi esetén – el kell pusztítania. A tereptárgyak, megvásárolhatók és lehelyezhetők, két hullám között és pénzbe kerülnek. A tereptárgyak akadályozhatják a játékos golyóit is, ezek a tereptárgyak nehezebben elpusztíthatók a zombik által, de a játékosnak is nehézséget okoznak.

- Pénz:

- Két hullám között a nyugalmi időben a játékosnak lehetősége van vásárolni a megszerzett pénzből. A játékos minden ölés után és megnyert hullám után pénzt szerez, ami egyenesen arányosan növekszik a játék nehézségével. A pénzt előre megtervezett módon érdemes elkölteni, hiszen ebből vásárolhatók:

- Tereptárgyak
- Fegyverek

- Lőszerek
 - Az el nem költött pénz megmarad a játékosnak.
- Játékos képességei
 - A játékosnak lehetősége van fejleszteni a képességeit a hullám végén megszerzett képességgel. Minden hullám után egy képességgel rendelkező pontot kap a játékos, amit mindenképp el kell használnia, hiszen a zombik erősödésével a játékosnak is fejlődnie kell. A képességek, amiket lehet fejleszteni:
 - Mozgási sebesség a pályán – Erre azért lehet szükség, mert a zombik is egyre gyorsabbak lesznek a játék előre haladásával
 - Lövési sebesség – Ezzel a képességgel a játékos gyorsabban, sűrűbben tud kilőni golyókat a kezében lévő fegyverrel, így könnyebben tudja legyőzni az ellenfeleket
 - Maximum életpont – Ezzel a képességgel több zombi támadást tud túlélni a játékos

Fejlesztői dokumentáció

Témaválasztás indoklása

A szakdolgozat témája már régebb óta foglalkoztatott. Az ilyen és ehhez hasonló, egyszerű, de megunhatatlan játékok nekem is tetszenek. Továbbá kíváncsi voltam miket lehet megvalósítani a WPF keretrendszerben, főleg a kijelzőről kilógó/túllógó elemek tekintetében.

Alkalmazott fejlesztői eszközök

A program `c#` programnyelvben, a WPF keretrendszert használva készítettem el. A .NET keretrendszer 4.7-es verziójával készült. A WPF keretrendszer jó választásnak bizonyult a beépített XAML leírónyelv miatt. Ezzel a nyelvvel könnyen, beépített módszerekkel lehet végezni a grafikai megoldásokat a programban. A XAML eszközei közé tartoznak a pályák kirajzolása, a pálya mozgatása, a pályaelemek mozgatása, a menü elemek kirajzolása és a többi grafikai megoldás is.

Használt programok

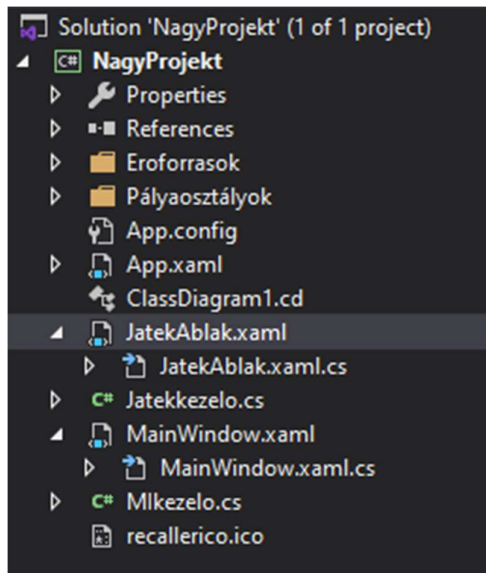
- Visual Studio 2019:
 - A fejlesztés teljes folyamata ebben történt meg, a `C#`, a XAML és a WPF használatához szükséges elemekkel rendelkezik a program.
- Adobe Photoshop:
 - A játékban található grafikák igazításához, méretezéséhez, átszínezéséhez, létrehozásához használtam ezt a programot. Pontos képszerkesztő, így könnyen lehet vele képeket alkotni.
- InstallForge:
 - A telepítőprogramot a játékhoz ezzel készítettem.

Adatmodell leírása

Programszerkezet

A program szerkezetét mappákba rendezve láthatjuk az ábrán, a programot több fő részre lehet bontani, ezek: erőforrások (képek, pályák), játékkezelő osztályok (a fő logikát valósítják meg), egyéb osztályok.

Az ábrán látható nem mappákba rendezett programelemek a játék fontosabb elemei, ezek:



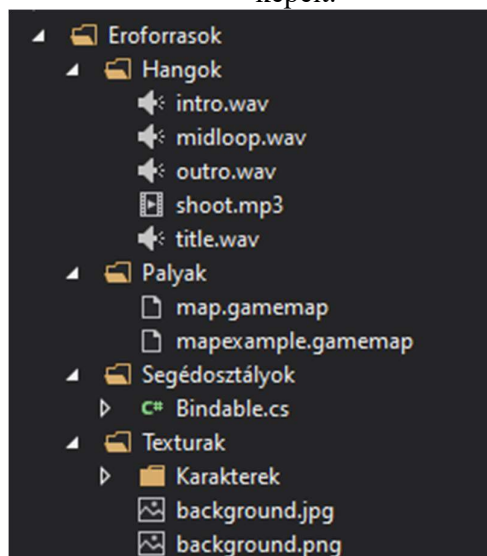
ábra 8 fájlok és mappák

- MainWindow.xaml
 - A fő menü kirajzolásáért felelős xaml
- MainWindow.xaml.cs
 - A fő menü mögötti metódusok megvalósításáért felelős, továbbá a fő menü gombjainak beállításáért
- JatekAblak.xaml
 - A játék ablak kirajzolásáért felelős xaml
- JatekAblak.xaml.cs
 - A játék ablaknak az adatainak beállításáért és átalakításáért felelős
- Jatekkezelo.cs
 - A játék teljes logikai irányításáért felelős irányító osztály, létrehozza a játék futtatásához szükséges változókat.
- MIkezelo.cs
 - A játékban lévő kezdetleges m.i. irányításáért felelős, pontos irányt ad a zombik mozgásának.

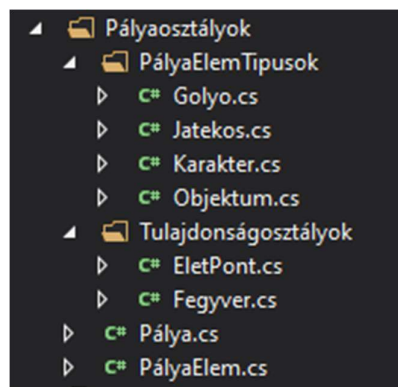
A következő ábrán a játék Erőforrások mappának a felépítése látható, ez olyan elemeket tartalmaz, amik a játék lecserélhető részei, kisebb mappákra van osztva:

- Hangok
 - A játék összes zenéjének, hangeffektjének tárolásáért felelős mappa.

- A hangok .wav illetve .mp3 formátumban vannak eltárolva a mappában.
- Pályák
 - A pályák egyszerű szövegesfájlként vannak eltárolva a mappában.
 - A pályafájlok kiterjesztése .gamemap.
- Segédosztályok
 - A Bindable segédosztály található ebben a mappában, ami a játék majdnem összes osztályának őssztálya.
- Textúrák
 - A játékban látható kép és grafikai elemek itt találhatóak.
 - A képek .png/.jpg/.jpeg formátumúak.
 - Ezen a mappán belül található egy karakterek mappa is:
 - Ennek a mappának a feladat eltárolni a különböző pályaelemek képeit.



ábra 9 erőforrások



ábra 10 pályaosztályok

A másik mappa, amivel találkozunk a programban az a PalyaOsztályok mappa, ez a mappa a játék fontosabb osztályait tárolja:

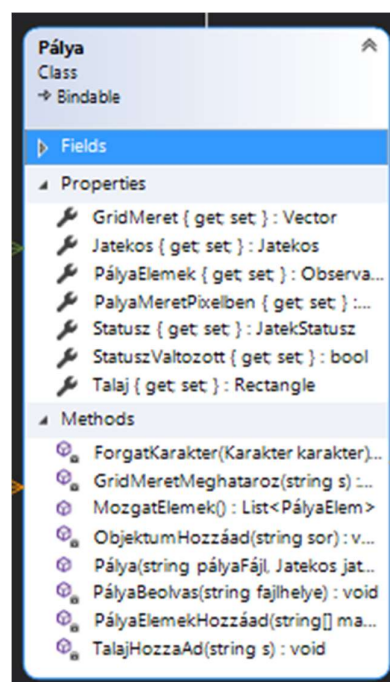
- Palya.cs
 - A pálya osztályt tartalmazza, legfontosabb feladata létrehozni a pályát és kezelni, mozgatni, eltárolni, törölni a rajta látható elemeket.
- PalyaElem.cs
 - A pályaelem osztályt tartalmazza, alapot képez a többi pályaElemnek, megadja a főbb tulajdonságokat, amivel minden pályaElem rendelkezik.
 - Ebbe az osztályba tartozik a pálya háttere/talaja.
- PalyaElemTipusok mappa
 - Karakter.cs
 - Speciális pályaElem ami képes mozgásra a pályán. Alapot képez a Golyó és Játékos osztályoknak. Ebben az osztályba tartoznak a zombik a játékban
 - Objektum.cs
 - Speciális pályaElem ami nem képes mozgásra a pályán. Ezzel lehet olyan elemeket hozzáadni a pályához amik akadályokat képeznek a pályán.
 - Golyo.cs

- Speciális pályaElem, azon belül speciális Karakter, ebbe tartoznak a pályán mozgó golyók, megkülönböztetési célból léteznek.
 - Jatekos.cs
 - Speciális Karakter, ez az osztály valósítja meg a Játékost a pályán. Továbbá ez az osztály tárolja a játékos fontos tulajdonságait.
- TulajdonságOsztályok mappa
 - Fegyver.cs
 - Ez az osztály a Játékos fegyvereinek a kezelésével foglalkozik.
 - EletPont.cs
 - Ez az osztály minden pályaElem életpontját kezeli, kezeli a pályaElem sebzését is.

Adatmodellek

A **pálya osztály** adatmodellje:

- Tulajdonságok:
 - GridMeret
 - Két adatból, egy x és egy y koordinátából álló vektor, a pályán szereplő elemeknek a szélességét és magasságát adja meg pixeleken.
 - Jatekos
 - A Jatekos osztályhoz tartozó adat, célja eltárolni a játékos tulajdonságait a pályának elérhető módon.
 - PályaElemek
 - PályaElem típusú, lista rendezett adatszerkezet, ami a pályán található összes pályaElem-et eltárolja és elérhetővé teszi külső osztályoknak.
 - PalyaMeretPixelben
 - Vektorként megadott x és y koordinátából áll. A pálya méretét tárolja el pixeleken megadva.
 - Statusz
 - Enumként megadott adat, eltárolja, hogy a játéknak épp mi a státusza.
 - StatuszValtozott
 - Bool típusú változó, célja jelezni külső osztályok felé, ha megváltozott a játék státusza
 - Talaj
 - Rectangle típusú változó, ami a pálya elemei alatt elhelyezkedő talajt tárolja el könnyen elérhető módon. A rectangle változó egy xaml-beépített kirajzolható elem.
 - MIKezelo

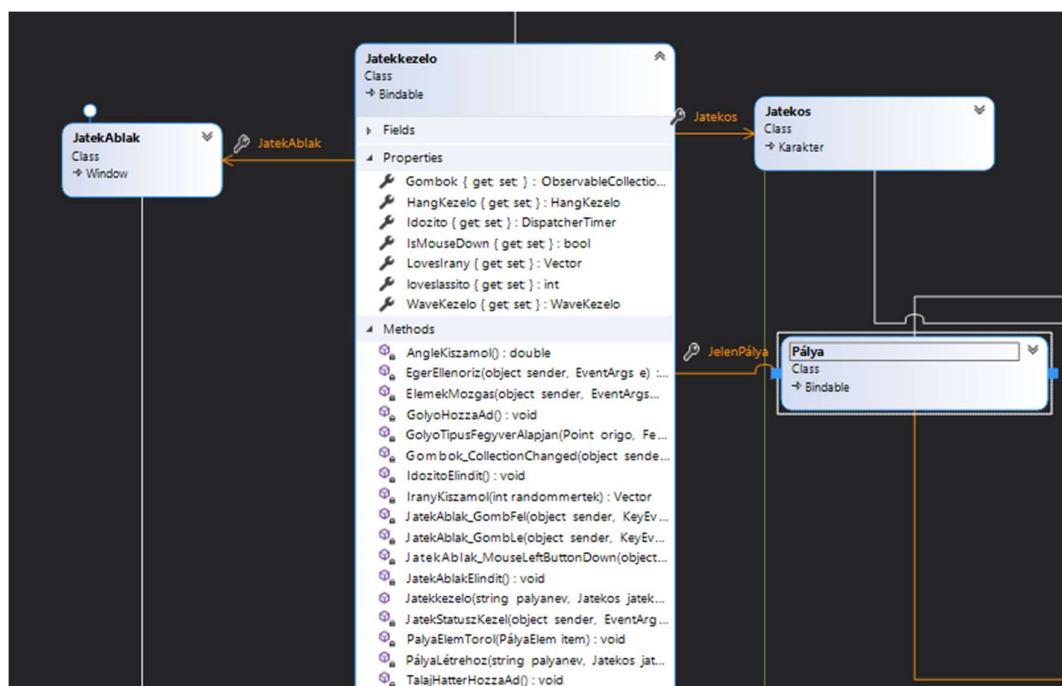


ábra 11 pálya osztály diagram

- MIKezelo típusú változó. A pályán lévő karakterek mozgását és „agyát” irányító osztály.
- Metódusok:
 - Pálya
 - A pálya konstruktora
 - PályaBeolvas
 - Az eljárás meghívja a TalajHozzaAd, GridMeretMeghataroz, PályaElemekHozzáad metódusokat és felbontja a pályafájl olyan módon, hogy a metódusok fel tudják dolgozni a szövegrészeket.
 - Forgatkarakter
 - Visszatérési értékkel rendelkező függvény, a karakter paraméternek a modelljét a karakter irány tulajdonsága alapján forgatja meg és adja vissza a megforgatott Rectangle-t (a karakter modellje ami kirajzolásra kerül)
 - PályaElemekHozzáad
 - Az eljárás hozzáadja a pályához a fájlból beolvasott objektumokat. Meghívja minden új objektumnál az ObjektumHozzáad metódust.
 - GridMeretMeghataroz
 - Az eljárás értéket ad az osztályban található GridMeret tulajdonságnak a megadott szöveg paraméter alapján, ami a beolvasott pálya fájl része.
 - **MozgatElemek**
 - A függvény meghívása esetén mozgatja a pályán található elemek összességét az adott pályaelem iránya és a pályán található többi elem függvényében. Visszatérési értéke egy lista azokról a pályaelemekről, amiket törölni kell a pályáról.
 - A függvény az egyik legfontosabb függvény a játékban hiszen ez minden pályaElem mozgását végzi és végez összehasonlításokat.
 - A függvény végigmegy egy ciklussal az egész JelenPálya listán és az alábbi ellenőrzéseket végrehajtja a vizsgált elemen mielőtt változtat a helyzetén:
 - Vizsgálja, hogy az adott pályaElem milyen típusú elem és el is tárolja a típusát a vizsgált elemnek, és ez alapján végzi a következő utasításokat; például a talaj sima pályaElem típusú így ezt nem lehet mozgatni, továbbá az objektumokat se lehet mozgatni.
 - Vizsgálja, hogy a pályaElem elhagyná-e a pályát, ha változtatnánk a helyzetén. Golyó esetén ez az elem törlését jelenti, míg Játékos vagy Karakter esetén a mozgás megakadályozását.
 - Egy újabb ciklussal végigmegy a játék többi elemén is és összehasonlítja mindegyik, nem azonos típusú elemmel, hogy az adott elem ütközne-e és ha ütközik, akkor milyen feladatokat kell végrehajtania:
 - Golyó-Objektum – ha az objektum golyoAtmegy tulajdonságának értéke igaz, akkor nem sebzi meg, egyéb esetben az Objektum a Golyó sebzésének értékét veszi az ÉletPontjából

- Golyó-Karakter – ha a golyó ütközik nem Játékos és nem Golyó típusú karakterrel, akkor a Karakter sebződik.
- Karakter-Jatekos – Ütközés esetén a Játékos és a Karakter is tud tovább mozogni, de a Játékos sebződik.
- Karakter-Objektum – Ütközés esetén az adott Karakter vagy Játékos nem tud továbbmenni. Nem Játékos karakter esetén az Objektumot sebzí a karakter.
- Karakter-Golyó – Ütközés esetén nem történik semmi, hiszen a sebzést a golyók végzik el ütközéskor
- ObjektumHozzaad
 - Az eljárás hozzáadja a pályához a fájlból beolvasott objektumot. Paramétere a megadott sora a fájlban.
- TalajHozzaAd
 - Az eljárás hozzáadja a pályához a fájlból beolvasott talajt és textúráját. A paramétere a pálya fájl talajra vonatkozó sorának a szövege.

A játékkezelő osztály adatmodellje:



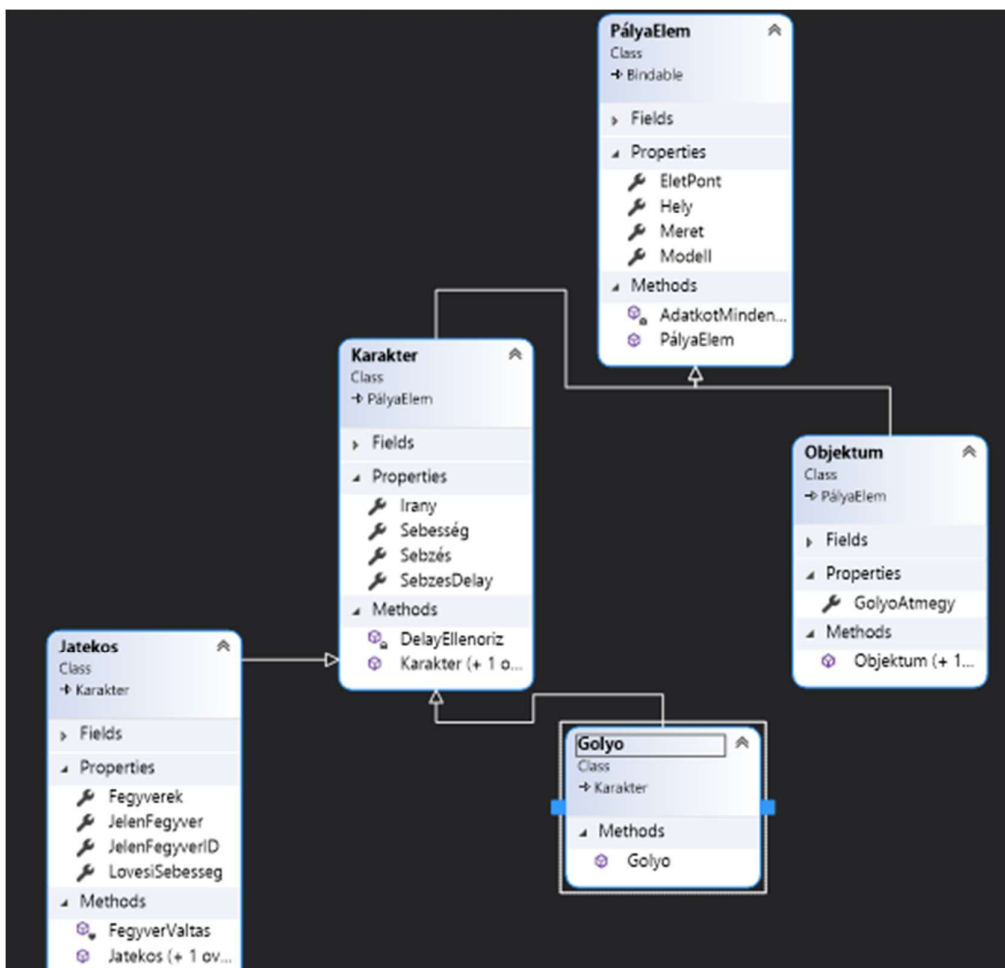
ábra 12 játékkezelő osztály diagram

- Tulajdonságok:
 - Gombok
 - Key enum értékeket tároló lista. Célja eltárolni a lenyomva tartott gombokat a játék folyamatos működése érdekében.
 - HangKezelo
 - HangKezelo típusú tulajdonság. Ez kezeli és tárolja a játékban játszott zenéket

- *Idozito*
 - DispatcherTimer típusú változó. Ez az időzítő hívja meg meghatározott időközönként, más folyamatokkal párhuzamosan a megadott eljárásokat.
- *IsMouseDown*
 - Bool típusú változó, igaz, ha az egérgomb le van nyomva
- *JatekAblak*
 - JatekAblak típusú speciális Window változó. A Jatekkezelo itt tárolja a megjelenített ablakot
- *Jatekos*
 - A Jatekkezelo a pályán kívül is tárolja a Játékos típusú játékost, hogy a tulajdonságait több pályán is lehessen használni.
- *JelenPálya*
 - Pálya típusú tulajdonság. Ez tárolja el a jelenleg játszott pályát és tulajdonságait a pályának.
- *LovesIrany*
 - Vector típusú változó. X és Y koordinátákból álló változó, ez a változó tárolja a célzás és lövés irányát a játékosnak relatíve a játékos pozíciójához képest.
- *Loveslassito*
 - Int típusú változó, ez számlálja és lassítja a golyók létrehozásának sebességét. Ha az értéke 0 akkor lő a játékos
- *WaveKezelo*
 - WaveKezelo típusú változó. Olyan változó ami tárolja és kezeli a hullámokat, továbbá kezeli a nyugalmi időszakokat a játékban.
- *Metódusok:*
 - *Jatekkezelo*
 - Az osztály konstruktora
 - *AngleKiszamol*
 - A függvény kiszámolja a szöveget, amiben a játékos lő a lövésirány alapján.
 - *EgerEllenoriz*
 - Olyan eljárás, ami adott időközönként ellenőrzi az egér pozícióját és állítja a játékos forgását. Emellett a LovesIranynak is értéket ad, továbbá meghívja a GolyoHozzaAd metódust, ha az egérgomb le van nyomva.
 - *ElemekMozgas*
 - Olyan eljárás, ami időközönként van meghívva és értéket ad a Jatekos Irányának. Továbbá meghívja a JelenPálya MozgatElemek metódusát, majd a mozgás után törli a visszatérő elemeket, Emellett mozgatja a kijelzőt, hogy a játékos a képernyő közepén legyen.
 - *JatekStatuszKezel*
 - Olyan eljárás, ami időközönként van meghívva és a játék státusza alapján küld eljárásokat a WaveKezelo-nek illetve JátékKezelo többi részének
 - WaveKezd-WaveMegy-WaveVege
 - Az adott hullámnak különböző szakaszai vannak, ahol a hullám elején kevés, a közepén sok, a végén

- kevés zombi jön megint. Ezek kontrolállják a zenét is a játékban
- Megallitva
 - Ha a játékos megállítja a játékot, akkor ez kezeli az időzítő megállítását
- Vasarlas
 - Adott hullám vége után következik be, ilyenkor nem lehet lőni, de lehet vásárolni és megjelenik a vásárlói menü is a játékban. Ellenfelek nem jönnek ránk.
- Elvesztes
 - Ha a játékos ÉletPontjának a JelenErteke eléri a 0-t a Játékos meghal, elveszíti a játékot, ez a státusz kezeli ezt az esetet.
- GolyoHozzaAd
 - Olyan eljárás, ami meghívás esetén létrehoz egy golyót és megadja a tulajdonságait; sebesség, irány, sebzés, stb., azzal, hogy meghívja a GolyoTipusFegyverAlapjan függvényt. Továbbá hozzáadja a golyót a JatekAblakhoz megjelenítésre, illetve a JelenPálya PályaElemekhez eltárolásra. Emellett a Jatekos adott fegyverének a lőszerét is kezeli.
- GolyoTipusFegyverAlapjan
 - Olyan függvény, ami visszaadja a Játékos jelenlegi fegyvere és annak tulajdonságai alapján, hogy milyen golyót kell létrehozni a pályára.
- Gombok_CollectionChanged
 - Olyan esemény, ami akkor kerül meghívásra, ha a Gombok lista tartalma megváltozik. A lista alapján meghívásra kerülnek a Játékos különböző metódusai, tulajdonságai. Például a FegyverValtas metódus, vagy a Jatekos Irany tulajdonsága kerül beállításra.
- IdozitoElindit
 - Olyan eljárás, ami értéket ad és feliratkoztatja a megfelelő metódusokat az időzítőre eseményként. Továbbá elindítja az időzítőt.
- IranyKiszamol
 - Olyan függvény, ami visszaadja a megadott paraméter alapján a irányt, amiben lőni fog a játékos. A megadott paraméter véletlenszerűvé teszi a lövés irányát a paraméternek megfelelően.
- JatekAblakElindit
 - Olyan eljárás ami az IdozitoElindit eljáráshoz hasonlóan a különböző eseményeket hozzáköti a JatekAblak eseményeihez. Ezek: JatekAblak_GombLe, JatekAblak_GombFel, JatekAblak_MouseLeftButtonDown. Továbbá megnyitja a JatekAblakot.
- *JatekAblak_GombFel*
 - Olyan esemény, ami gomb felengedésénél lesz meghívva. Az adott gombot eltávolítja a Gombok listából.
- *JatekAblak_GombLe*

- Olyan esemény, ami gomb felengedésénél lesz meghívva. Az adott gombot hozzáadja a Gombok listához.
 - *JatekAblak_MouseLeftButtonDown*
 - Olyan esemény, ami a bal egérgomb lenyomva tartása esetén van meghívva. A GolyoHozzaAd eljárást hívja meg.
 - *PalyaElemTorol*
 - Olyan eljárás, ami az átadott paraméterben szereplő pályaElemt törli a Pálya PályaElemek közül, továbbá eltávolítja a JatekAblakról is a hozzá tartozó modellt. Mindezt a pályaElem típusához megfelelően teszi.
 - *PalyaLetrehoz*
 - Olyan eljárás, ami értéket ad a JelenPályának, továbbá beállítja a Jatekos Helyét.
 - *TalajHatterHozzaAd*
 - Hozzáadja a JátékAblakhoz a JelenPálya talaját.
- PályaElem, Karakter és Jatekos osztályok adatmodellje:**



ábra 13 pályaelem és származtatott osztályok diagramja

- PályaElem:
 - Tulajdonságok:
 - Eletpont

- EletPont típusú tulajdonság. Minden pályaelemnek van életpont tulajdonsága. Ez határozza meg, mennyi életpontja van az adott karakternek. Van egy MaxErtek, aminél több nem lehet sose az életpont és egy JelenErteke mindkettőt double-ként tárolja a program.
 - Hely
 - Vector típusú tulajdonság. Helye minden pályaelemnek van, a Vector X és Y koordinátát tárol el a pályán, ezzel meghatározva a helyüket. Ez hozzá van kötve a modellhez.
 - Meret
 - Vector típusú tulajdonság. Mérete minden pályaelemnek van, a vector x és y koordinátája a szélességét (x) és magasságát (y) adja meg az elemnek pixelben. Hozzá van kötve a modellhez.
 - Modell
 - Rectangle típusú tulajdonság. Ez egy olyan típus amit ki tud rajzolni a WPF és ennek a Rectanglenek a pozíciója és mérete hozzá van kötve a PályaElem adataihoz. Ezt adjuk hozzá a WPF ablakhoz, hogy kirajzolásra kerüljön az adott pályaelem, továbbá ezzel vizsgáljuk a pályaelem ütközését.
 - Metódusok:
 - AdatkotMindenAdat
 - Ez az eljárás a Modell tulajdonságait köti a PályaElem többi tulajdonságához. A méretét, helyét itt adjuk át neki.
- Karakter:
 - Tulajdonságok:
 - Irany
 - Vector típusú tulajdonság. A benne lévő x és y értéke 0 és 1 között mozog. A játékos helyéhez adódik hozzá az iránynak az X és Y koordinátája. Például: Irány = 0,1 Hely = 0,0 esetén, ha hozzáadjuk az irányt a helyhez a Hely = 0,1 lesz.
 - Sebesség
 - Double típusú tulajdonság. Feladata eltárolni azt a szorzót amivel az irányt megszorozzuk minden mozgásnál, értéke 0-nál nagyobb bármekkora szám lehet. Például: Irány = 1,1 Sebesség = 1.2 esetén az új érték 1.2,1.2 lesz.
 - Sebzés
 - Double típusú tulajdonság. Feladata eltárolni azt az értéket, amivel meg tud sebezni más pályaelemeket. Amikor egy golyó eltalál egy zombit is ezt használjuk, illetve mikor a zombi eltalálja a játékost vagy a pályaelemeket is ezt használjuk. Értéke 0-nál nagyobb bármekkora szám lehet.
 - SebzésDelay
 - Int típusú tulajdonság célja egy számlálót vezetni annak érdekében, hogy az adott karakter ne tudjon minden alkalommal sebezni. Működése közben, amikor 100 a SebzésDelay az adott karakter tud sebezni, viszont ha a karakter sebez, ezt beállítjuk egy kisebb értékre, ami

időközönként csökken. Amikor eléri, a 100-t újra tud sebezni. Alapértelmezett értéke 100.

- Metódusok:
 - DelayEllenoriz
 - A SebzesDelay értékadása során hívjuk meg, ha a SebzesDelay eléri a 0-t akkor visszaállítjuk 100-ra a SebzesDelayt.
- Jatekos:
 - Tulajdonságok:
 - Fegyverek
 - Lista típusú tulajdonság, ami fegyver típusú elemeket tárol. Lényege a játékos összes fegyverének, lőszerének és a fegyverek tulajdonságainak eltárolása
 - JelenFegyver
 - Fegyver típusú tulajdonság, ami a jelenleg használt fegyver tulajdonságait tárolja el.
 - JelenFegyverID
 - Int típusú tulajdonság, azt az indexet tárolja el, ami most használt fegyver indexe
 - LovesiSebessegSzorzo
 - Double típusú tulajdonság, ami nevéből adódóan is egy szorzó. Ez legalább 1-es alapértelmezett értékkel rendelkezik, de tovább növelhető, maximum 2-ig.
 - Metódusok:
 - FegyverValtas
 - Ennek az eljárásnak a meghívása esetén a váltás számával megfelelően növeljük, vagy csökkentjük a JelenFegyverID-t. Ha az ID nagyobb vagy kisebb mint a lista mérete, akkor megfelelően változtatjuk az ID. Nagyobb = 0-ra csökkentjük. Kisebb = Lista utolsó indexére állítjuk be.

Tesztelési dokumentáció

1. Teszt:
 - a. Teszteset: Megpróbáltam a fő menüben rávinni az egerem egy elemre miközben egy másik már ki volt jelölve.
 - b. Elvárt eredmény: A másik elem nem kapja meg a kijelölés effektet.
 - c. Kapott eredmény: A teszt sikeres volt, a kapott eredmény megegyezett az elvárásokkal
2. Teszt:
 - a. Teszteset: Megpróbáltam kimenni a pályáról, a játékost a pálya széléhez mozgatva
 - b. Elvárt eredmény: A játékos megakad a pálya szélénél, nem tud továbbmenni.
 - c. Kapott eredmény: A teszt sikeres volt, a kapott eredmény megegyezett az elvárásokkal
3. Teszt:
 - a. Teszteset: Megpróbáltam átírni a játék Erőforrásaiban a pályafájlt.
 - b. Elvárt eredmény: A felhasználó nem tudja átírni a pályafájlt.
 - c. Kapott eredmény: A teszt sikertelen volt, a pályafájlt tudja módosítani a felhasználó.
4. Teszt:
 - a. Teszteset: Megpróbáltam átmozogni egy pályaelemen a pályán, miután már elpusztították a zombik
 - b. Elvárt eredmény: Az elem eltűnik és a helyén szabadon lehet mozogni.
 - c. Kapott eredmény: A teszt sikeres volt, a karakterek át tudnak mozogni a törölt pályaelem helyén.
5. Teszt:
 - a. Teszteset: Megpróbáltam a lövés gomb illetve a mozgás gombok nyomkodásával elérni, hogy gyorsabban lőjön vagy mozogjon a karakter.
 - b. Elvárt eredmény: A lövés és a mozgás gyorsaságán ennek nem szabad változtatnia, továbbra is a korábbi sebességgel kell mindkettőnek történnie.
 - c. Kapott eredmény: A teszt sikeres volt, a dolgok sebessége nem változik a különböző gombok váltakozó nyomásától.

Továbbfejlesztési lehetőségek

Történet üzemmód

A jelenlegi állapotában a játék elég egyhangú és hasonlít a régebbi „rogue-like” játékokhoz. Vagyis a játékos, ha elveszíti a játékot, kezdheti a legelejéről újra. Ezen szeretnék javítani azzal, hogy a játékba egy történet módot teszek, amiben különböző pályákon át ugyanúgy hullámokra osztva nehezebb és nehezebb feladatokat kell elvégezni a játékosnak.

Grafikai fejlesztések

A játék grafikájának jó része külső forrásból van, ez egyszerűség számomra, de így nem eléggé testre szabható az élmény. Ezen szeretnék javítani azzal, hogy esetleg új textúrákat rajzolok a karaktereknek, golyóknak. Ezen kívül a mozgás és a lövés is rendelkezhetne hozzájuk illő animációval. Továbbá lehetnének a játékban olyan robbanó/tűz effektek amiket a fegyverek tüzelésével lehetne kiváltani.

Változatos ellenségek

Az ellenfelek a jelenlegi formában egyféleképpen tudják közelíteni és támadni a játékost. Ezt szeretném kiegészíteni olyan módon, hogy jobb M.I-t és bővebb karaktertípust adok a játéknak. Előbbit egy útkereső algoritmussal lehetne kiegészíteni, hiszen ezek az algoritmusok a mai modern játékok alapvető elemei. Ezt az algoritmust akár be is lehetne állítani, hogy bekerítse a játékost, vagy kövesse, vagy várjon egy olyan pillanatra, mikor több zombival egyszerre tud támadni a játékos ellen. Utóbbit pedig úgy lehetne bővíteni, hogy esetleg olyan támadási lehetőségeket adunk a zombiknak, amivel távolabbról vagy falakon keresztül is tudnak támadni, de cserébe nagyon gyengék. Továbbá fő-ellenség szerű zombit is lehetne betenni, amit például 5 hullám után le kell győznie a játékosnak, hogy tovább juthasson. Ez a zombi magasabb életponttal, nagyobb sebesséssel, de cserébe lassú sebességgel rendelkezhetne.

Játékszabályok kiegyensúlyozása

A játékszabályok kiegyensúlyozása alatt azt értem, hogy lehetne változtatni azon, mennyi pénzt kap a játékos, milyen drágák a fegyverek, barikádok és mennyit fejlesztenek a játékoson a különböző tulajdonságok növelései. Továbbá folyamatos tesztelésnek kéne kitenni a játékot, hogy meg lehessen figyelni, hogy adott stratégiákkal meddig lehet eljutni a játékban. A kiegyensúlyozáshoz még érdemes lehetne „örtornyokat” beletenni a játékba, amik automatikusan tüzelnek a zombikra, de cserébe kevés életük van és nagyon drágák. Ezeknek beépítése is érdekes lehet.

Ábrajegyzék

ábra 1 nyelv választás	3
ábra 2	3
ábra 3	4
ábra 4	4
ábra 5 játékos tulajdonságai	6
ábra 6 golyók, tereptárgyak, zombi	7
ábra 7 vásárló menü	8
ábra 8 fájlok és mappák	12
ábra 9 erőforrások	13
ábra 10 pályaosztályok	13
ábra 11 pálya osztály diagram	14
ábra 12 játékkezelő osztály diagram	16
ábra 13 pályaelem és származtatott osztályok diagramja	19

Irodalomjegyzék, forrásmegjelölés

[1] Zombi és túlélő képek:

<https://opengameart.org/>

Utolsó elérés: 2021.04. 15

[2] Optimalizációs problémák megoldása:

<https://stackoverflow.com/>

Utolsó elérés: 2021. 04. 15.

[3]Zenék:

<https://soundcloud.com/lilfm/sets/recallergameost>

Utolsó elérés: 2021. 04. 15.

[4]Osztály diagram létrehozása:

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

Utolsó elérés: 2021. 04. 15.

[5]Telepítő létrehozása:

<https://codedocu.com/>

Utolsó elérés: 2021. 04. 15.

[6]Lövés hangeffektek:

<https://www.fesliyanstudios.com/>

Utolsó elérés: 2021. 04. 15.