ZÁRÓDOLGOZAT

Amghar-Szabó Dániel

2021

**Kocka Hősök**

**Amghar-Szabó Dániel**

**SZÁMALK-Szalézi Szakgimnázium**

**Szoftverfejlesztő**

**Konzulens: Rózsás Adrienn**

**Nyilatkozat**

a záródolgozat eredetiségéről

Alulírott Amghar-Szabó Dániel (név) {Szabó Ildikó Éva (anyja neve) 386218TA (szem. ig. szám)} büntetőjogi és fegyelmi felelősségem tudatában kijelentem és aláírásommal igazolom, hogy a záródolgozat saját munkám eredménye. A felhasznált irodalmi és egyéb információs forrásokat az előírásoknak megfelelően kezeltem, a záródolgozat készítésre vonatkozó szabályokat betartottam.

Kijelentem, hogy ahol mások eredményeit, szavait vagy gondolatait idéztem, azt a záródolgozatomban minden esetben, beazonosítható módon feltüntettem, a dolgozatban található fotók és ábrák közlésével pedig mások szerzői jogait nem sértem.

Kijelentem, hogy a záródolgozatom elektronikus változata teljes egészében megegyezik a nyomtatott formával.

Hozzájárulok ahhoz, hogy az érvényben lévő jogszabályok és a Számalk-Szalézi Szakgimnázium belső szabályzata alapján az iskola saját könyvtárában megtekinthető (olvasható) legyen a záródolgozatom.

A záródolgozat titkos/nem titkos.

Budapest, 2019. április 06.

Amghar-Szabó Dániel

Tanuló aláírása

Tartalomjegyzék

[1 Bevezetés 4](#_Toc68635193)

[1.1 Miért választottam a témát: 4](#_Toc68635194)

[1.2 A játék alapötlete: 4](#_Toc68635195)

[2 Felhasználói dokumentáció 6](#_Toc68635196)

[2.1 Rendszer követelmények 6](#_Toc68635197)

[2.2 Telepítési útmutató 6](#_Toc68635198)

[2.3 Menüpontok 6](#_Toc68635199)

[2.3.1 Töltőképernyő 6](#_Toc68635200)

[2.3.2 Főmenü részei 7](#_Toc68635201)

[2.3.3 Regisztráció 7](#_Toc68635202)

[2.3.4 Belépés 8](#_Toc68635203)

[2.3.5 Beállítások 8](#_Toc68635204)

[2.3.6 Pontok 9](#_Toc68635205)

[2.3.7 Világválasztó 9](#_Toc68635206)

[2.3.8 Pályaválasztó 10](#_Toc68635207)

[2.4 A játék leírása 10](#_Toc68635208)

[2.4.1 Irányítás 10](#_Toc68635209)

[2.4.2 Ellenfelek 11](#_Toc68635210)

[2.4.3 Pontok 11](#_Toc68635211)

[2.4.4 Pályacélja 11](#_Toc68635212)

[2.4.5 Pályák 12](#_Toc68635213)

[3 Fejlesztői dokumentáció 13](#_Toc68635214)

[3.1 Fejlesztői környezet 13](#_Toc68635215)

[3.1.1 Unity 13](#_Toc68635216)

[3.1.2 Visual Studio 2019 13](#_Toc68635217)

[3.1.3 Apache NetBeans 13](#_Toc68635218)

[3.1.4 XAMPP 13](#_Toc68635219)

[3.2 Felhasznált modulok 13](#_Toc68635220)

[3.2.1 Simplejson 13](#_Toc68635221)

[3.2.2 Spriteok 13](#_Toc68635222)

[3.2.3 Betűtípus 14](#_Toc68635223)

[3.2.4 Hangok 14](#_Toc68635224)

[3.3 Adatbázis 14](#_Toc68635225)

[3.3.1 Szerkezet 14](#_Toc68635226)

[3.3.2 Funkciók 15](#_Toc68635227)

[3.4 Osztályok 15](#_Toc68635228)

[3.4.1 Unity 15](#_Toc68635229)

[3.4.2 PHP 17](#_Toc68635230)

[3.5 Tesztelés 22](#_Toc68635231)

[3.6 Továbbfejlesztési lehetőségek 28](#_Toc68635232)

[3.6.1 Játékélmény 28](#_Toc68635233)

[Játékélmény 28](#_Toc68635234)

[3.6.2 Monetizálás 28](#_Toc68635235)

[4 Összefoglalás 30](#_Toc68635236)

[5 Felhasznált anyagok 31](#_Toc68635237)

[5.1 Fileok 31](#_Toc68635238)

[6 Irodalomjegyzék 32](#_Toc68635239)

Ábrajegyzék

[ábra 1:Töltőképernyő 6](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665641)

[ábra 2: Főmenü 7](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665642)

[ábra 3:Regisztráció 7](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665643)

[ábra 4:Belépés 8](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665644)

[ábra 5:Beállítások 8](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665645)

[ábra 6:Pontok 9](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665646)

[ábra 7:Pontok 2 9](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665647)

[ábra 8:Világválasztó 10](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665648)

[ábra 9:Pályaválasztó 10](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665649)

[ábra 10:Karakter irányitás 11](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665650)

[ábra 11: Ellenfél 12](#_Toc68665651)

[ábra 12:Pontszámok kijelezve 13](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665652)

[ábra 13:Adatbázis táblák 16](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665653)

[ábra 14:Belépés php kód 20](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665654)

[ábra 15:Regisztráció php kód 20](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665655)

[ábra 16:Pont megszerzése php kód 21](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665656)

[ábra 17:Pontok mentése php kód 22](file:///C:\Users\win7\Downloads\AmgharSzaboDaniel_Záródolgozat%20korr2.docx#_Toc68665657)

# Bevezetés

## Miért választottam a témát:

A videójátékokban rengeteg lehetőség rejlik, és engem is érdekelnek a videójátékok. Emiatt szakdolgozatom témájául egy videójáték készítését választottam. Így belepillanthatok a játék készítés világába. Érdekesnek találom a játékos adatainak mentési folyamatát szerveroldalról, az ellenség mesterséges intelligencia gyártását, és a pálya készítést is. Rengeteg témában és stílusban lehet videójátékot készíteni. Sok különböző ötlet jutott az eszembe, végül egy platformerre esett a választásom.

Platformerek között is rengeteg különbség van. Például vannak a fizika alapú platformerek, amikben különböző fizikai objektumokat tudunk tologatni a pályán való tovább jutáshoz; vagy a kirakós típusúak, amikben különböző elemeket kell a megfelelő helyre rakni a következő pályára való jutáshoz. Én végül egy ügyességen alapuló mászkálós játék mellet döntöttem.

Ezután a platform kiválasztásáról kellett döntenem. Végül a mobil platfom típust választottam. Választásom fő oka, hogy a mobil játékok nagy népszerűségnek örvendenek. A mobil piacon rengeteg a lehetőség az áttörésre, hatalmas potenciál rejlik ezen a területen. A legnagyobb piaccal rendelkező mobil platform jelenleg az Android, ezért arra fogom gyártani a játékot.

Végül a játékmotorról (game engine-röl) kellet döntenem, ebben több különböző szempontot is figyelembe kellet vennem: a kezelhetőséget, és a kompatibilitást mobiltelefonokkal. Végül eldöntöttem, hogy Unity game engine-ben készítem el. Választásom oka a hordozhatóság: Unityban könnyedén át konvertálhatom a projektemet más platformokra a későbbiekben.

## A játék alapötlete:

A Kocka Hősök alapkoncepciója, hogy van egy játékosunk, akinek az a feladata, hogy eljusson A pontból B pontba, különböző akadályokat elkerülve. Ezek lehetnek sebző felületek vagy ellenséges egységek. A felhasználónak lehetősége van kisebb jutalmakat megszereznie a játék során, például aranyérméket és egyéb különböző jutalmakat.

# Felhasználói dokumentáció

## Rendszer követelmények

A játék futtatásához egy Android rendszert futtató érintő képernyővel rendelkező okostelefonra van szükség. A játék teljes élményének átadásához a telefon képernyőjének a felbontása minimum 1920x1080 pixelnek kell lennie. Az eredmények mentéséhez folyamatos internet kapcsolatra van. A jelenlegi teszt felhasználó adatok a [teszt@gmail.com](mailto:teszt@gmail.com) email cím és az 123 jelszóval lehet bejelentkezni.

## Telepítési útmutató

A játék telepitéséhez a mobil telefonunkra kell felvinni a KockaHosok.apk nevű filet utána meg kell keresni a telefonnak a fájlkezelőében. A file megtalálása után rá kell koppintani a telepítőnek az ikonjára. Az ikonra való koppintás után rákérdez, hogy ismeretlen forrásból lévő alkalmazást akar telepíteni beleegyezünk utána telepítőik az alkalmazás.

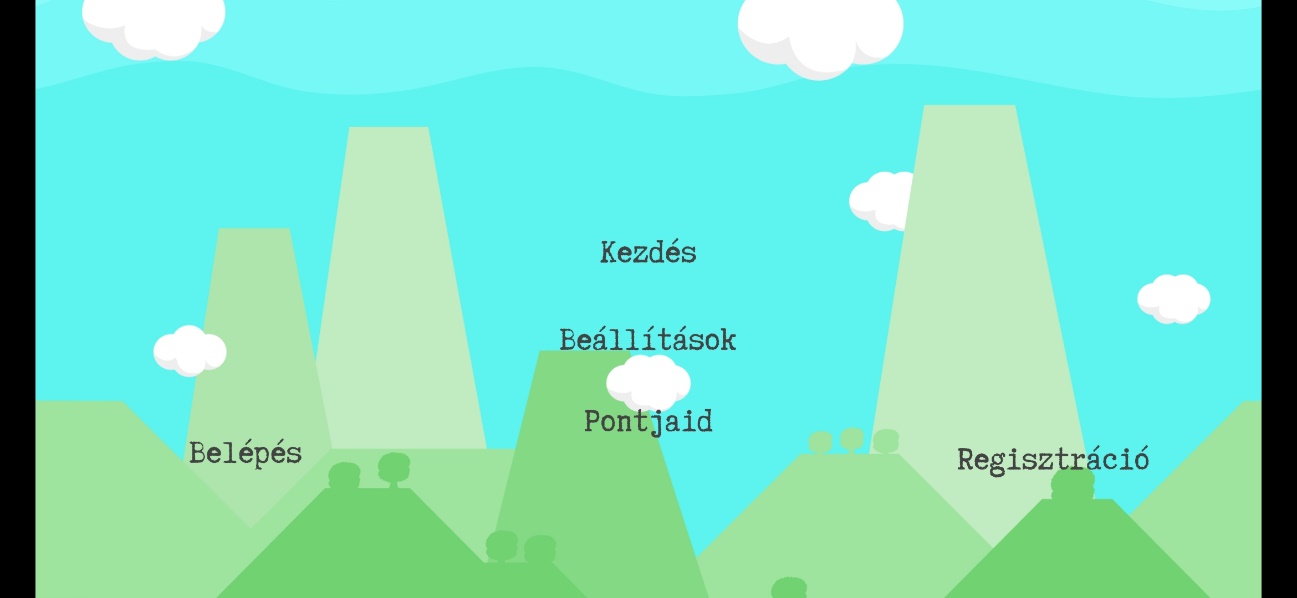
## Menüpontok

### Töltőképernyő

A játék elindításakor egy töltő képernyőt láthatunk három másodpercig, utána tovább jutunk a főmenübe.

ábra :Töltőképernyő

### Főmenü részei

Az első almenü a *Belépés* gomb, amire kattintva lehetősége van a már regisztrált felhasználónak belépnie a játékba. A jobb alsó sarokban foglal helyet a *Regisztráció* almenü, amire kattintva elkezdhetjük kitölteni a belépéshez szükséges regisztrációs folyamatot. A következő almenü a *Beállítások*, ahol a játék beállításait végezük. A *Beállítások* alatt egy *Pontjaid* almenüt láthatunk, ahol az elért pontok olvashatóak. Ha szeretnénk elindítani a játékot, akkor az első, *Kezdés* gombra kell kattintani, amivel eljutnunk a világválasztóba.

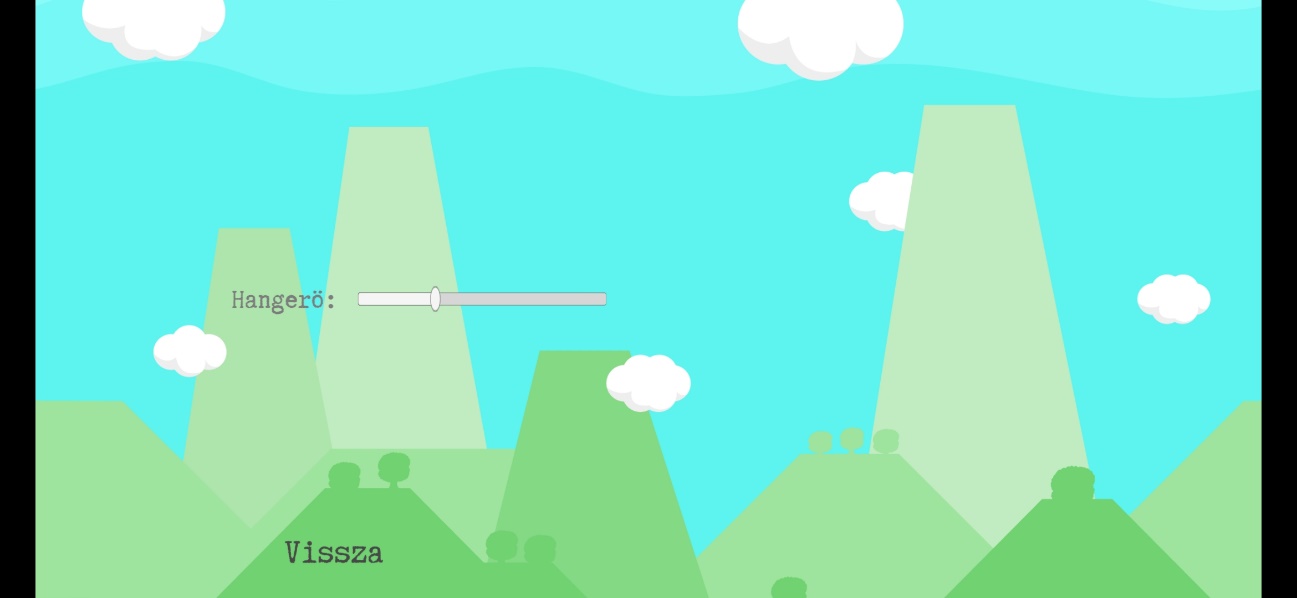
ábra : Főmenü

### Regisztráció

Itt az első rubrikába egy, még nem használt e-mail címünket kell megadni. Ezután meg kell írnunk egy tetszőleges felhasználónevet, majd ezt követően a jelszót, kétszer is. A két jelszó mező tartalmának meg kell egyeznie a sikeres regisztráció érdekében. Ha minden mezőt sikeresen kitöltöttünk, akkor a *Regisztráció* gombra kattintva tudunk beregisztrálni. Található a felületen egy *Vissza* gomb, aminek segítségével vissza tudunk lépni a fő menübe.

ábra :Regisztráció

### Belépés

A bejelentkezéshez szükségünk van egy érvényes e-mail cím és egy jelszó megadására. Ha sikeresen megadtuk az adatainkat a *Belépés* gombra kattintva tudunk bejelentkezni. A jobb alsó sarokban láthatunk egy *Vissza* gombot, aminek segítségével lehetőségünk van visszakerülni a fő menübe.

ábra :Belépés

ábra :Beállítások

### Beállítások

A *Beállítások* menüben, egy hangerőszabályzó csúszka található, ahol a játék hangerejét tudjuk állitani. Innen is a *Vissza* gomb segítségével tudunk visszajutni a főmenübe

### Pontok

A *Pontok* menüben a bejelentkezett felhasználó jelenlegi legmagasabb pontjait olvashatjuk. Amennyiben még nem vagyunk bejelentkezve, egy a bejelentkezésre felhívó rövid üzenetet fogunk látni.

ábra :Pontok



ábra :Pontok 2

### Világválasztó

A világválasztó menüben egy kiválasztható világot láthatunk. Erre kattintva a pályaválasztóba jutunk. Ezenkívül a jobb felső sarokban található egy *Menü* gomb, aminek segítségével visszakerülünk a főmenübe.



ábra :Világválasztó

### Pályaválasztó

A pályaválasztón belül hat különböző pályát láthatunk, ezekből válogathatunk. Ezek egyikére rákattintva elindul a játék a már előzetesen kiválasztott pályáról. A jobb felső sarokban szintén találunk egy *Menü* gombot, tehát bármikor, a játék menete közben lehetőségünk van arra, hogy visszalépjünk a főmenübe.

ábra :Pályaválasztó

## A játék leírása

### Irányítás

A karakterünk irányítása egy kontroller segítségével történik, ami csak akkor látható, amikor az ujjunkat a telefon kijelzőjéhez érintjük az ujjunk ballra és jobbra mozgatásával tudjuk a karakterünk mozgatni balra és jobbra. A képernyő jobb alsó sarkában egy piros körre kattintva tud a karakterünk ugrani. Az imént felsorolt karaktermozgás-irányítására egyszerre is van lehetőség. Ez azt jelenti, hogy ha a karakterünkkel át akarunk ugrani egy akadályt, akkor egyszerre fogjuk az ugrás gombot és a kontrollert alkalmazni.

ábra :Karakter irányitás

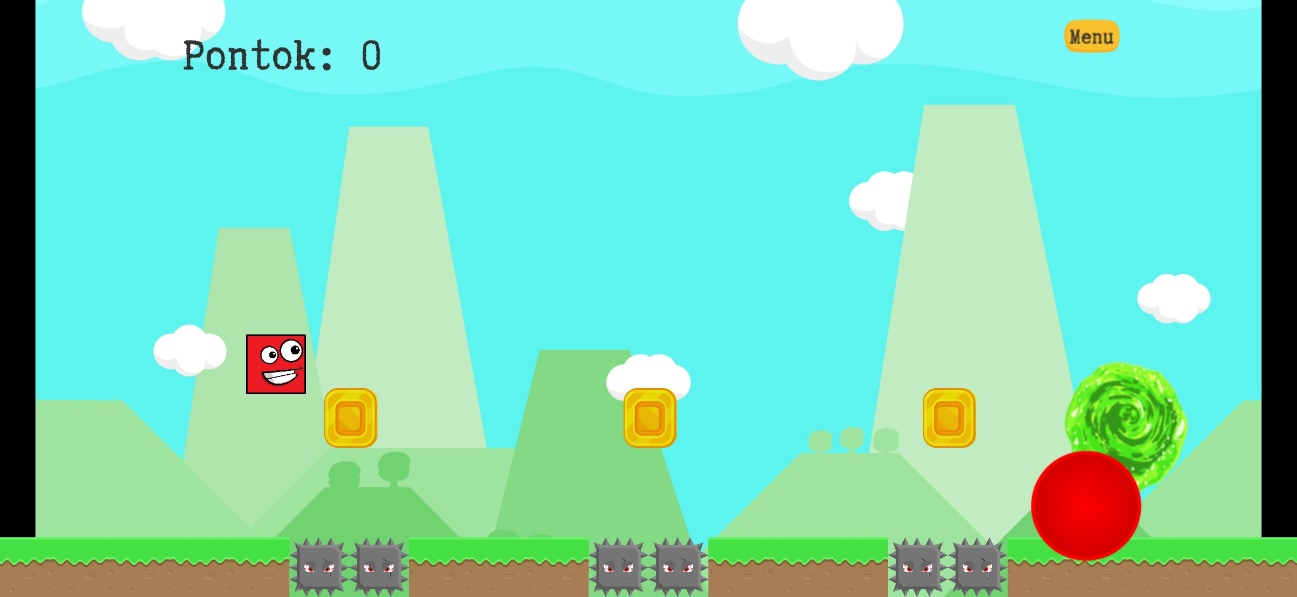
### Ellenfelek

A pályán tüskével vannak jelezve az akadályok, amiket a karakterünknek el kell kerülnie a tovább jutás érdekében. A pályákon találhatóak még lyukak, amiket a karakterünknek át kell ugrania. Későbbi pályákon megjelenik egy másik akadály, ami egy mozgó vörös kocka képében látható, ami jobbra-balra irányuló mozgásban folyamatosan csúszkál a pályán. A karakterünknek ezeken is át kell ugrania a sikeres továbbjutás érdekében. Bármelyik akadályba csúszunk bele, a játék automatikusan visszadob az aktuális pálya elejére.



ábra :Ellenfél

ábra :Lyuk



ábra :Tüskék

### Pontok

A pályákon különböző helyeken pontokat gyűjthet a felhasználó. Ezek a pontok aranyérmék formájában láthatók. Megszerezhetőtek, ha a karakterünk megérinti őket. Az arany érméken kívül azzal is képes a karakterünk pontot szerezni, ha ráugrik ellenségek fejére. Minden pálya végén, ha bevagyunk jelentkezve, és van internet kapcsolatunk, a pontjaink lementésre kerülnek. Ha jobb az előzőleg megszerzett pontértéknél, akkor az felülírja a már meglévőt. Az első játék esetén egyszerűen lementi a megszerzett pontokat. A pályán lévő pontjainkat a bal felső sarokban elhelyezett pontszámlálón láthatjuk.



ábra :Érme

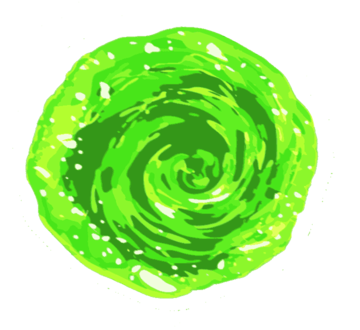
ábra :Pontszámok kijelezve

### Pályacélja

Minden pályán van egy végpont, ami egy zöld portál. A pálya akkor ér véget, ha a karakterünk túljutott minden akadályon, és megérinti a zöld portált. Ilyenkor tovább navigál automatikusan a következő pályára. Az utolsó pálya a világon kivétel, ilyenkor a felhasználó átkerül a *Pontjaid* menübe, ahol láthatók az eddig elért pontok.



ábra :Pontok menü



ábra :Pája cél

### Pályák

Miután elindította felhasználó a játékot, a karakter a bal alsó sarokban fog állni egy piros négyzet alakjában. Az első pályán az ugrás ikonja felett látható a pálya végét jelző zöld portál, ahova el akarunk jutni. A hármas számú pálya érdekes, mert egy kisebb lépcsőszerű szerkezeten kell felugrálnia a karakterünknek, majd egy nagyobb lendületű ugrás szükséges ahhoz, hogy a szakadék túloldalára kerüljünk, ahol pedig egy lefelé vezető lépcsős szerkezeten juthatunk tovább.

Ahogyan haladunk előre a pályákkal, a nehézségi szint is emelkedik. Az ötödik pályán már plusz egy szinten is végig kell mennie a karakterünknek, hogy sikeresen kijuthasson. Ehhez egy zöld négyzetre kell felugrania, majd onnan a második szintre. Itt azonban ügyeskednünk kell, mert a négyzet mögötti hosszú, barna falban fennakadhat a karakter, de onnan könnyedén visszajuthatunk a kontrollerrel.

A hatodik, utolsó pályán már kettő vörös mozgó akadály nehezíti a továbbjutást, kiegészítve szakadékokkal, illetve egy megszakított második szinttel is, ahol az egyik mozgó akadály található. Az utolsó pályák már sokkal nagyobb koncentrációt és odafigyelést igényelnek.

# Fejlesztői dokumentáció

## Fejlesztői környezet

### Unity

A Unity-nak a Personal kiadását használtam abból a 2019.4.21.f1-es vertióját a játék fejlesztéséhez. A Unity adja meg a fejlesztés középpontját, ebben a játék engine-ben készültek el a játékosok és a pályák főbb elemei. Ebben a környezetben futtatam és teszteltem a játékot az Unity remote segítségével. Az Unity remote egy kiegészítő, ami segít mobiltelefonos környezetben tesztelni a játékot.

### Visual Studio 2019

A Visual Studionak a community változatából a 16.6.3 verziójú IDE-t használtam. A Visual Studioban az Unityban létrehozott karakterek és elemeknek a scriptjeit segített módosítani.

### Apache NetBeans

Az Apache NetBeans 12.0 verzióját használtam. Az NetBeans IDE-ben a szerverrel való kapcsolat létrehozását segítő fileok elkészítésében segített.

### XAMPP

A XAMPP nak a 3.2.4es verzióját választottam. XAMPP segítségével sikerült létrehoznom az adatbázist, amivel a NetBeans-ben létrehozott php fileok segítségével sikerült tovább küldenem a játékba az adatokat.

## Felhasznált modulok

### Simplejson

A Simplejson egy modul, amivel a szerverről érkező string típusú, de már jsonné formált adatokat segít átdolgozni egy json objektummá, amiból könnyebb kiolvasni a beérkezett adatokat. A modult a FelhasználoPontok.cs nevű fileban Pontok kiír metódusban a lekérdezésből beérkezett adatok feldolgozására használom fel.

### Spriteok

A játék hátteréhez és a talaj elemeihez felhasznált spriteok valamint a tüskékre a BayatGames Free Platform Game Assetseket használtam ezenkívül a fő képernyőhöz a játékos karakteréhez és az ellenfél karakteréhez Pallagi Bence által készített spriteokat használtam a pálya végét jelző portálhoz egy portálról egy gifet alakítottam át spriteokká.

### Betűtípus

Mivel a játéknak az alapvető betűtípusok közül nem találtam semmi megfelelőt ezért a menükben lévő betűtípust a dafont nevű weboldalon kerestem. A <https://www.dafontfree.net/freefonts-truetypewriter-polyglott-f102970.html> url alatt találtam egy játékhoz illőt.

### Hangok

A játékban lévő hangokat a freesoundon nevű weboldalról töltöttem le. A háttérzenét a <https://freesound.org/people/Mrthenoronha/sounds/522297/> a töltő képernyőn szereplő hangot a <https://freesound.org/people/original_sound/sounds/494979/> linken találtam.

## Adatbázis

Az adatbázisban a jelenleg regisztrált játékosokat és a létező pályákat tároljuk. Ha a felhasználónk elért egy pontszámot a pályánkon, akkor felveszük az elért pontszámát pályától függően a felhasználónak.

### Szerkezet

A felhasznalok táblában a felhasználónak az email címét felhasználó nevét és a jelszavát hashelve tárolom.

A palyak táblában a pályáknak az értékeit tárolom.

A ranglista táblában a felhasználóknak az elért pontszámát tárolom pályánkként.

**felhasznalok** (email, felhnev, jelszo)

**palyak** (palyaid, vilag, palya)

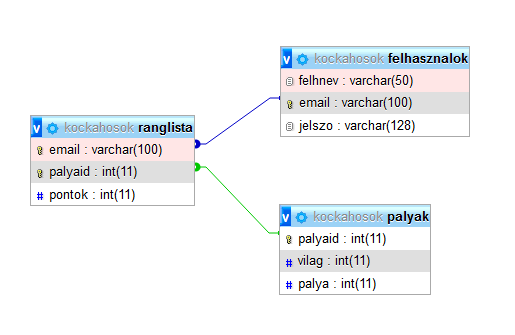
**ranglista** (email, palyaid, pontok)

**felhasznalok:**

|  |  |  |
| --- | --- | --- |
| **Oszlopnév** | **adattípus** | **Szerep** |
| email | varchar (100) | kulcs, külsőkulcs |
| felhnev | varchar (50) |  |
| jelszo | varchar (128) |  |

**palyak:**

|  |  |  |
| --- | --- | --- |
| **Oszlopnév** | **adattípus** | **Szerep** |
| palyaid | int (11) | kulcs, külsőkulcs |
| vilag | int (11) |  |
| palya | int (11) |  |

**ranglista:**

ábra :Adatbázis táblák

|  |  |  |
| --- | --- | --- |
| **Oszlopnév** | **adattípus** | **Szerep** |
| email | varchar (100) | kulcs |
| palyaid | int (11) | kulcs |
| pontok | int (11) |  |

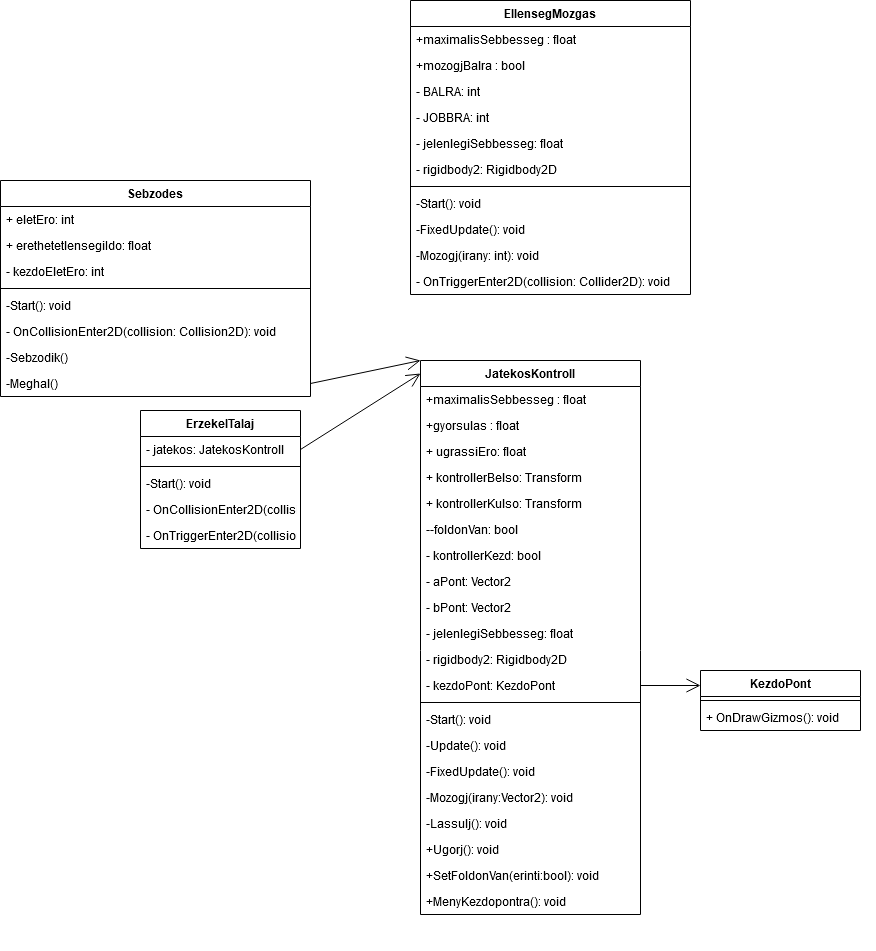
### Funkciók

* A felhasználó emailcímének, jelszavának és felhasználónevének felvétele
* Új pályák felvétele
* Felhasználó által elért pontszám felvitele a megfelelő pálya mellé
* A felhasználó által elért pontszámnak a frissítése

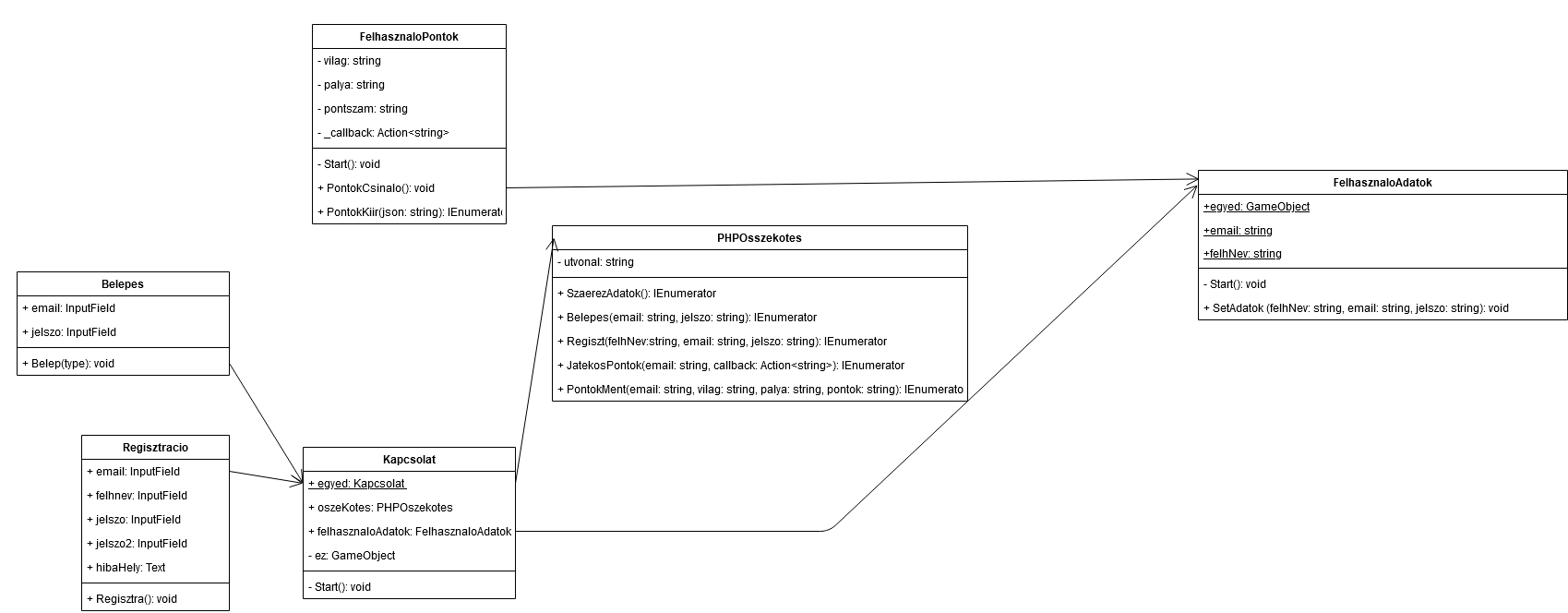
## Osztályok

Ebben a fejezetben az általam elkészített osztályok működését fogom leírni.

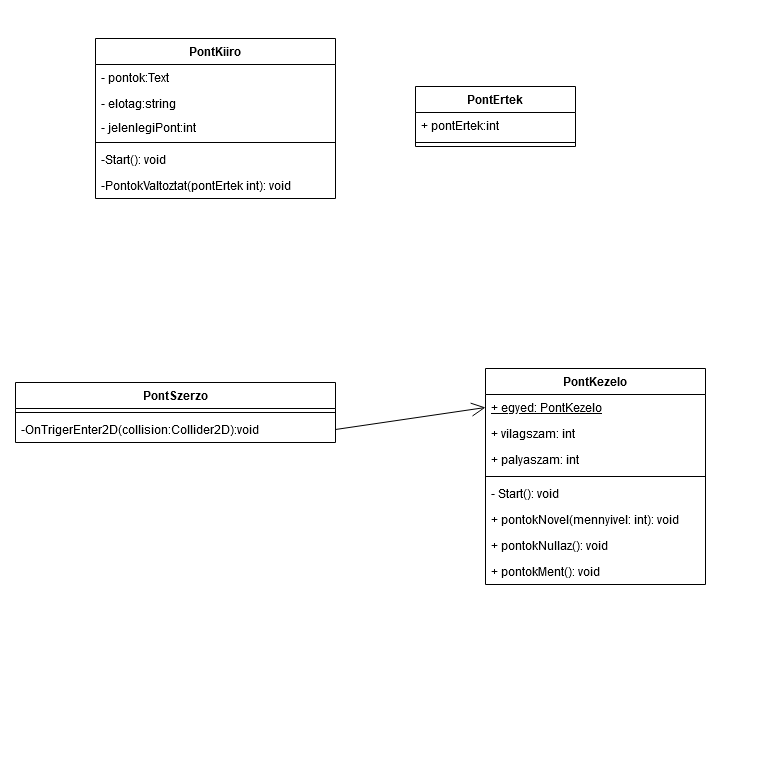
### UML ábrák

Az alábbi ábrákat a játékban lévő különböző osztályok kapcsolatait mutatják be.

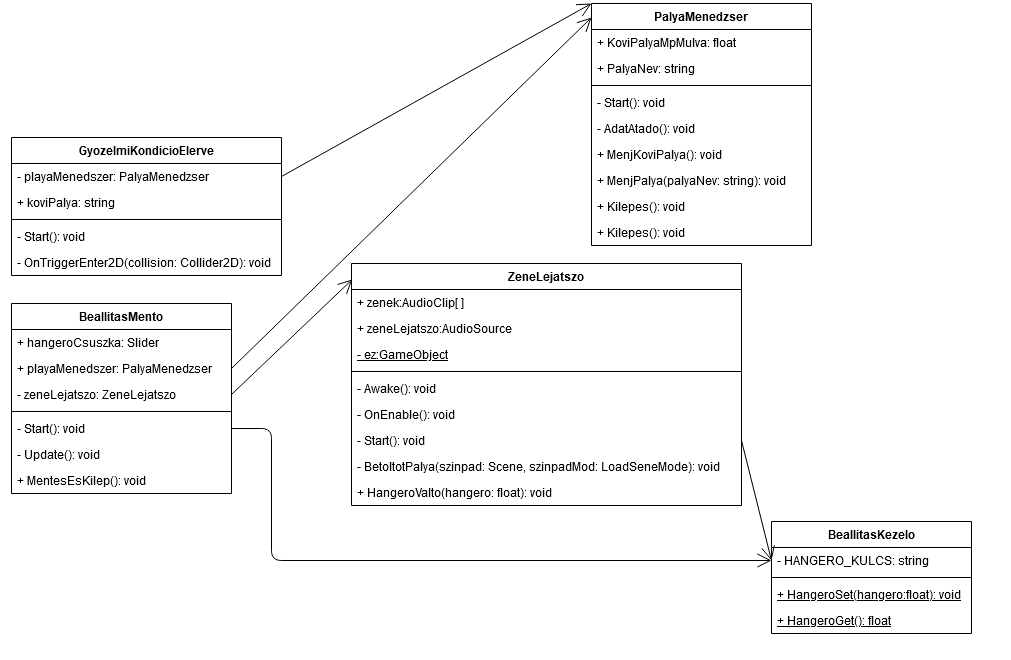
ábra 18: UML ábra 1



ábra 19:UML ábra 2



ábra 20UML ábra 3



ábra 21:UML ábra 4

### Unity

**JatekosKontroll**

A JatekosKontroll osztály a játékos objektumra van ráhelyezve és l megadott paraméterek alapján mozgatja a karaktert. És inicializál a beállított spriteokkal egy kontrollert, ha a játékos megérinti a képernyője bal oldalát.

Publikus változók

* maximalisSebesseg – Az objektum mozgási sebességének a maximumát alítja
* gyorsulas – Az objektum sebesség növekedésének mennyiségét álltija.
* ugrassiEro - Az objektum ugrásának a nagyságát alítja.
* kontrollerKulso – A kontroller hátsó spriteját tárolja
* kontrollerBelso - A kontroller mozgó spriteját tárolja

Fontosabb privát változók

* foldonVan – Azt tárolja hogy a játékos érinti e a földet
* kezdoPont – Egy kezdoPont típusú objektumot tárolunk benne

Metódusok:

Mozogj(Vector2 irany):A játékos objektumnak a pozícióján változtat x dimenzióban

Lassulj():A játékos sebességén csökkent és pozícióján változtat

Ugorj(): Ha a játékosnak a foldonVan értéke true akkor a játékos objektumunknak az y kordinátán ad erőt.

Pattanj():A játékos objektumunknak az y koordinátán ad erőt.

SetFoldonVan(bool erinti) :A foldonVan változónak az értékét lehet beállítani vele

MenyKezdopontra(): A kezdőpont változónak a pozíciójára transzformálja a játékos objektumunkat

**Kapcsolat**

Ezen az osztályon keresztül lehet elérni a PHPOszekotetes nevű osztályt.

Publikus változók

* egyed – Az osztálynak egy statikus példánya
* oszekotes – Egy PHPOszekotes típusú változó private settere van
* **felhasznaloAdatok** – FelhasznaloAdatok típusú változó privát szettere van

Metódusok:

Start(): Ebben a metódusban inicializálóik az oszekotes a felhasznaloAdatok és az egyed változó.

**FelhasznaloAdatok**

Ebben az osztályban tároljuk a felhasználó adatai ezeket az adatokat a statikus egyedén át érjük el majd az osztálynak.

Publikus változók

* egyed – Az osztálynak egy statikus példánya
* email – A játékosnak az email címét tárolja privát settere van.
* felhNev – A játékosnak a felhasználó nevét tárolja privát settere van

Metódusok:

SetAdatok(string felhNev,string email, string jelszo) :ebbe a metódusban tudjuk beállítani a változókat

**PontKezelo**

A PontKezelo osztályban

Publikus változók

* egyed – Az osztálynak egy statikus példánya
* pontok – A pontok értékét tároljuk benne privát settere van a változónak
* pályaszám – A pályának a számát tárolja
* világszám – A világnak a számát tárolja

Metódusok

pontokNovel(int menyivel) :Ezzel a metódussal változtatjuk meg az összes pontszámnak az értékét.

pontokNullaz():Az elért pontok értékét nullázza le.

pontokMent():Az elért pontszámot mentjük le a Kapcsolat statikus egyede használatával

### PHP

**Kapcsolat Osztály**

Változók mind privátak

* $hoszt – A szerver nevét tárolja
* $fnev -A szerver felhasználó nevét tárolja
* $jelszo - A szerver jelszavát tárolja
* $adatbNev – Az adatbázis nevét tárolja
* $kapcsolat - A konstruktor által létrehozott kapcsolat objektumot tárolja

**Metódusok**

\_\_construct ()***:*** Az adatbázis kapcsolatot hozza létre és menti le a $kapcsolat nevű változóba

adatBeszuras ($tablanev, $oszlopNevek, $adatok)***:*** A $tablanev nevű táblában illeszti be az adatokat. A $oszlopNevek paramétertömbből kiolvassa azokat az oszlopneveket, ahova beakarunk szúrni. A $adatok tömbből pedig a hozzátartozó értékeket olvassa be, amiket be akarunk illeszteni a sorba. Ha sikeres a beszúrás, akkor kiírja, hogy 'sikeres beszúrás', ha nem sikerül, akkor, hogy 'sikertelen beszúrás '.

keresesOszlopban ($tablanev, $oszlopNev, $mit): A $tablanev nevű táblából keresi a $oszlopNev nevű oszlopban a $mit nevű adatot.

keresesTablaban ($tablanev, $mit): Az egész $tablanev elnevezésű táblában keresi a $mit elnevezésű adatot, Ha talál, akkor kiírja, hogy 'van elem a táblában', ha nincs akkor ‘nincs elem a táblában', rossz adatok esetén, hogy ‘hibás lekérdezés'.

keresesAdatokTablaban ($tablanev, $sorok, $miket): ugyanolyan, mint a keresesOszlopban nevű metódus csak ebben adott sorban tudunk adott adatot keresni visszaadja az adott sorokat

adatTorles ($tablanev, $kulcsnev, $kulcsertek): táblából töröl egy sort, ahol a $kulcsnev értéke az egyenlő a $kulcsertek nevű változóval; ha sikerült a törlés, kiírja, hogy 'sikeres törlés ' ha nem akkor 'sikertelen törlés '.

adatModositas ($tablanev, $kulcsnev, $kulcsertek, $oszlopnev, $mivel): táblában módosít adatot egyszerre csak 1 adatot tud módosítani és egy értéken vizsgálja, hogy megegyezik-e a kulcs név és az érték

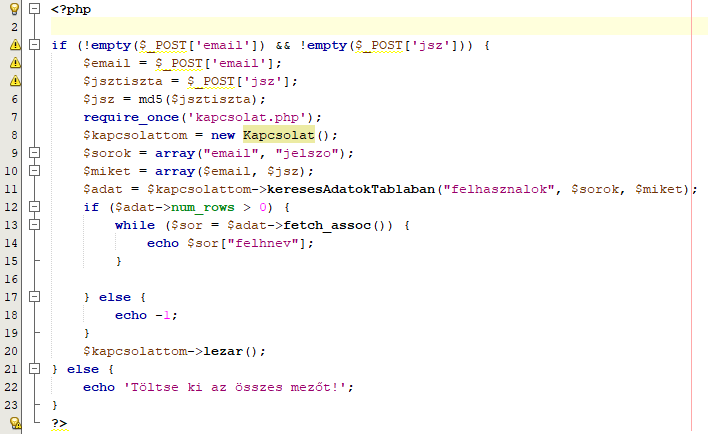
adatModositasOszetet ($tablanev, $kulcsnev, $kulcsertek, $oszlopnev, $mivel):

ugyan olyan, mint az adatModositas metódus, csak itt több $kulcsnev és $kulcsertek is tudunk vizsgálni, mivel arraynek kell lenni.

lekerdezMind($tabla): ez egy extra funkció, amivel könnyen lehet megtekinteni az adatbázisban szereplő adatokat visszaküldi az összes adatot egy táblából.

lezar (): lezárja az adatbázis kapcsolatot.

Ezeket a metódusokat meghívom a belepes.php, regisztracio.php, felhasznalopontok.php és a pontokMent.php fileokban. Ezekre a kódokra lépünk be a Unity felületéről és itt a különböző beérkező adatokat feldolgozzuk.

A belepes.php ban egy $\_POST globális tömb értékben szerepelnie kell az emailnek és egy másikban a jelszónak. A jelszavakat itt hasheljük, md5() php parancs segítségével. Ellenőrizzem, hogy szerepel-e felhasználó az adatbázisban, ha igen, visszaküld a felhasználó nevet, ha nem, akkor -1-et.

ábra :Belépés php kód

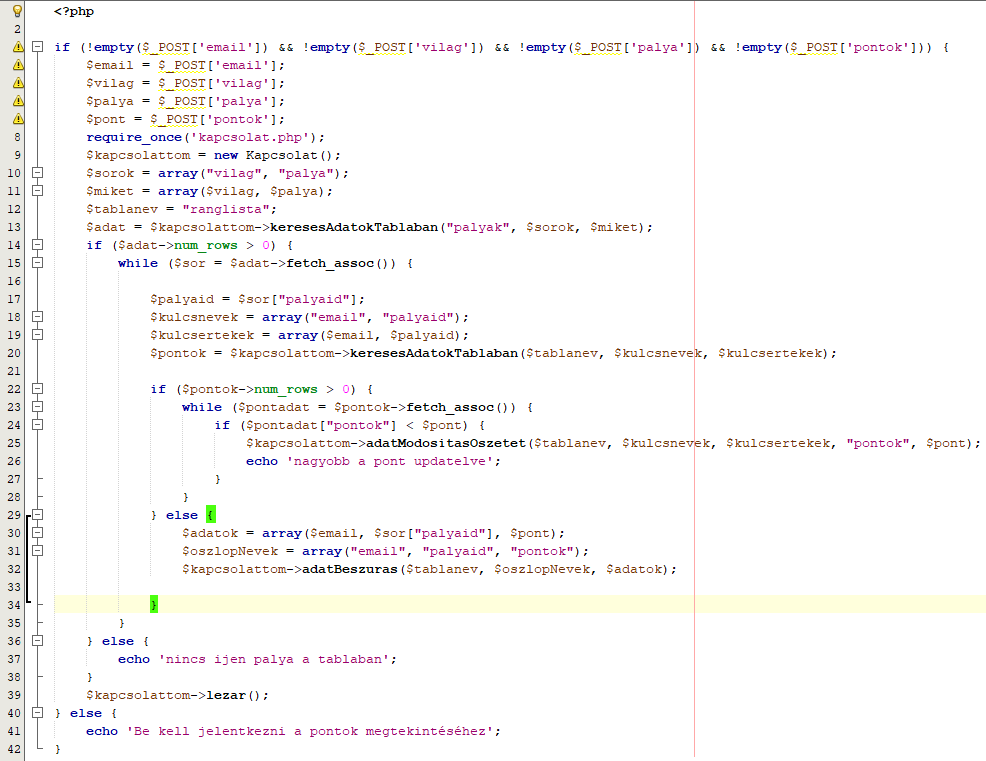
A regisztracio.php ban a $\_POST értékekben fel kell töltenünk felhasználó nevet egy jelszót és egy emailt a jelszót md5 en hasheljük, utána ellenőrizzük, hogy létezik-e az adatbázisban és utána, ha még nincs, felöltjük. Ha van kiírjuk, hogy van már ilyen felhasználó.

ábra :Regisztráció php kód

A felhasznalopontok.php fájlnak a használatához egy email címet kell megadni. Ezt megkeresünk az adatbázisban; ha létezik, akkor megkeressük a ranglistában szereplő személyek közül az ilyen email című felhasználókat és visszaadjuk; ha nincs ilyen email, akkor nem küldünk vissza semmit.

ábra :Pont megszerzése php kód

A pontokMent.php fájlnál meg kell adnunk egy email címet, egy világ számot, egy pálya számot és egy pontok számot. Ha létezik pálya és világ, akkor lekérdezzük az idejét és utána megkeressük a ranglista táblában az emaillel is, illetve a pálya idővel rendelkező táblát. Ha szerepel ilyen, megnézzük a pontszámot. Ha nagyobb a beérkezett pontszám, akkor frissítjük. Ha nincs ilyen elem azzal a pálya idővel és emaillel, akkor új sort illesztünk be a táblába.



ábra :Pontok mentése php kód

## Tesztelés

|  |  |  |
| --- | --- | --- |
| Teszteset | Elvárás | Eredmény |
| **Regisztráció tesztelése** | | |
| A felhasználó megfelelően regisztrál | A felhasználó adatok felkerülnek az adatbázisba | Sikeres teszt |
| Jelszó1 nem egyezik a jelszó 2 vel | Hiba lesz kijelezve a felhasználónak nem lesz felvéve az adatbázisba | Sikeres teszt |
| A felhasználó nem tölti ki a jelszót | Hiba lesz kijelezve a felhasználónak nem lesz felvéve az adatbázisba | Sikeres teszt |
| A felhasználó nem tölti ki a felhasználó nevet | Hiba lesz kijelezve a felhasználónak nem lesz felvéve az adatbázisba | Sikeres teszt |
| A felhasználó nem adott meg email | Hiba lesz kijelezve a felhasználónak nem lesz felvéve az adatbázisba | Sikeres teszt |
| A felhasználó létező emailt add meg | Nem lesz felvéve új felhasználó az adatbázisba | Nem vesz fel új felhasználót, de nem jelzi, hogy létezik ezzel az emaillel felhasználó |
| A felhasználó korábban felhasznált adatokkal akar regisztrálni | Nem lesz új felhasználó létrehozva | Nem lesz új sor, ha az email megegyezik már létezővel |
| A felhasználó megpróbál regisztrálni internet kapcsolat nélkül | Nem jön létre új felhasználó | Sikeres teszt |
| **Belépés tesztelése** | | |
| Létező adatokkal belép | Belépteti a rendszer | Sikeres teszt |
| A felhasználó megpróbál bejelentkezni üres mezőkel | Nem lépettibe a rendszer | Sikeres teszt |
| A felhasználó megpróbál bejelentkezni adatbázisban nemlétező email címmel. | Nem lépettibe a rendszer | Sikeres teszt |
| A felhasználó megpróbál bejelentkezni helyes emailcímmel de helytelen jelszóval | Nem sikerül a belépés | Sikeres teszt |
| A felhasználó megpróbál bejelentkezni helyes emaillel, de kitöltetlen jelszóval. | Nem sikeres a belépés | Sikeres teszt |
| A felhasználó megpróbál bejelentkezni internet kapcsolat nélkül | Nem sikerül a belépés | Sikeres teszt |
| **Pontok** | | |
| A felhasználó belép a pontokba bejelentkezés nélkül | Feliratot kap, ami felszólítja a bejelentkezésre | Sikeres teszt |
| A felhasználó belép a pontokba bejelentkezve | Kiírja az eddig elért pontjait | Sikeres teszt |
| A felhasználó belép a pontokba anélkül, hogy valaha is játszott volna | Nem ír ki semmit | Sikeres teszt |
| A felhasználó belép a pontokba anélkül, hogy minden pályán végig megy | Kiírja az eddig elért pontjait | Sikeres teszt |
| A felhasználó belép a pontokba, hogy minden pályán végig ment | Kiírja a minden pályán elért pontokat | Sikeres teszt |
| A felhasználó belép a pontokba, hogy végig ment pályákon többször | Az eddig elért legjobb pontjait írja ki. | Sikeres teszt |
| A felhasználó sikeresen bejelentkezett de elment az internet kapcsolat amikor belép a pontokba | Nem ír ki semmit. | Sikeres teszt |
| **Navigáció tesztelése** | | |
| Főmenüben a Kezdés gombra kattintás | Elkerülünk a világválasztó menübe | Sikeres teszt |
| Főmenüben a Beállítások gombra kattintás | Elkerülünk a beállításokba | Sikeres teszt |
| Főmenüben a Pontjaid gombra kattintás | Elkerülünk a pontok menüpontba | Sikeres teszt |
| Főmenüben a Belépés gombra kattintás | Elkerülünk a belépés menübe | Sikeres teszt |
| Főmenüben a Regisztráció gombra kattintás | Elkerülünk a regisztráció menübe | Sikeres teszt |
| Beállításokban a Vissza gombra kattintás | Elkerülünk a fő menübe. | Sikeres teszt |
| A világ választóban a Menü gombra kattintás | Elkerülünk a menübe | Sikeres teszt |
| A világ választóban a Világ1 gombra kattintás | Elkerülünk a | Sikeres teszt |
| A Belépés menüben a Vissza gombra kattintás | Elkerülünk a fő menübe. | Sikeres teszt |
| A Regisztráció menüben a Vissza gombra kattintás | Elkerülünk a fő menübe. | Sikeres teszt |
| A Pályaválasztó menüben a Menü gombra kattintás | Elkerülünk a menübe | Sikeres teszt |
| A Pályaválasztó menüben a pálya gombokra kattintás 1-6ig gombra kattintás | Elkerülünk a megfelelő pályákra | Sikeres teszt |
| Pályákon a Menü gombokra kattintás | Elkerülünk a menübe | Sikeres teszt |
| **Karakter irányitás tesztelése** | | |
| Ugrás gomb megnyomása | A karakterünk ugrik | Sikeres teszt |
| Képernyő baloldalára való kattintás | Megjeleni a játékost irányitó joystick | Sikeres teszt |
| Képernyő jobb való kattintás | Nem történik semmi | Sikeres teszt |
| Joystick elhúzása balra | A karakterünk mozdul balra | Sikeres teszt |
| Joystick elhúzása jobbra | A karakterünk mozdul jobbra | Sikeres teszt |
| Ugrás gomb gyors megnyomása egymás után | Karakterünk egyszer ugrik | Néha a karakterünk nagyobbra ugrik a számítottnál |
| Ugrásgomb lenyomva tartva | Ugrik egyet és nem történik semmi | Sikeres teszt |
| Képernyő jobb oldala lenyomása közben megnyomjuk a balt. | A kontrollert elő hozzuk és ugrunk egyet | Sikeres teszt |
| **Ellenségek Tesztelése** | | |
| Halálos talaj objektummal való érintkezés | A játékosunk a talaj objektummal való érintkezés után vissza kerül a kiinduló pontjára | Sikeres teszt |
| Halálos mozgó objektummal való érintkezés | A játékosunk a mozgó objektummal való érintkezés után vissza kerül a kiinduló pontjára | Sikeres teszt |
| Pályán lévő lyukba való esés | A játékosunk a lyukba való esés után vissza kerül a kiinduló pontjára | Sikeres teszt |
| **Pontok mentésének tesztelése** | | |
| Először megy végig a pályán a felhasználó 0 pontot ér el | Felkerül az adatbázisba 0 ponttal | Nem viszi fel az adatbázisba a 0pontot |
| Először megy végig a pályán a felhasználó több mint 0 pontot ér el | Új sor kerül az adatbázisba az elért pontszámmal | Sikeres teszt |
| Másodszor megy végig a pályán a felhasználó és megint nem ér el pontot | Nem változik az elért pontszám. | Nem viszi fel az adatbázisba a 0 pontot a második alkalommal sem. |
| Másodszor megy végig a pályán a felhasználó kevesebb mint az előzőleg elértpontot ér el | Nem változik az elért pontszám. | Sikeres teszt |
| Másodszor megy végig a pályán a felhasználó ugyan annyi pontot ér el mint előzőleg | Nem változik az elért pontszám. | Sikeres teszt |
| Másodszor megy végig a pályán a felhasználó több pontot ér el mint előzőleg | Az előző elért pontszám táblát frissítjük az új pontszámmal. | Sikeres teszt |
| Belépés nélkül megy végig a felhasználó a pályán | Nem lesz adatbázisban változás | Sikeres teszt |
| **Fehér dobozos tesztelés** | | |
| FelhasznaloPontok class PontokKiir metódusába nem json formátumú adatot kap | Hibát add vissza |  |
| FelhasznaloPontok class PontokKiir metódusába json formátumú adatot kap, de van a végén extra adat | A végén lévő adatokat nem rakja be a jsonbe | Sikeres teszt |
| FelhasznaloPontok class PontokKiir metódusába megfelelő adatot kap | Sikerrel legyártja az objektumot | Sikeres teszt |
| A PHPOszekotetes class ban a Regiszt metódusban nem 3 adatot adunk meg | No arguemt errort kapunk | Sikeres teszt |
| A PHPOszekotetes class ban a Regiszt metódusban rossz típusú adatot adunk meg | Cannot convert errort kapunk | Sikeres teszt |
| A PHPOszekotetes class ban a Regiszt metódusban megfelelően adjuk meg az adatokat | Sikeresen felveszi az adatbázisba az adatokat | Sikeres teszt |

## Továbbfejlesztési lehetőségek

Rengeteg módon lehet tovább fejleszteni a játékot: a játék élményt, a játéknak a monotonitás elkerülését, gazdasági bevételét kiemelten.

### Játékélmény

A játékban jelenleg elég kevés pálya van. További világok elkészítésével és újabb pályák implementálásával meglehetne hosszabbítani a játéknak az idő tartamát.

Amennyiben megvan a textúrája a karakternek és az ellenségnek, animációt is lehetne készíteni hozzá. Amikor a karakter egyhelyben áll, egy várakozó animációt játszana le, és amint végig játszotta az animáció összes képkockáját, onnan elkezdné elölről. Amikor a karaktert éppen irányítjuk, akkor futás animáció lenne az idle helyett. Lenne az ugrásnak is animációja. Az ellenségnek egy sétáló animációja lenne. Az animáció több textúra egymás utáni lejátszásából készülne el.

A beléptetési rendszernél több vissza jelzést kellene kapnia a felhasználónak, és a regisztrációnál is ha sikeres.

### Monetizálás

A játék jelenleg érméket csak pontszerzési okból tud gyűjteni. A játékosok nincsenek annyira ösztönözve, hogy kitérjenek az útból egy nehezebb érme megszerzésért. Ezért az érme rendszert ki lehetne egészíteni azzal, hogy a pályán létező érméket, amit, ha a játékos összegyűjt, akkor kap játékbéli pénzt, amit a játékos képes lenne kiváltani jutalmakra.

Játékbéli bolt egy olyan funkció, amit, ha hozzáteszünk a játékhoz, akkor képesek leszünk monitalizálni a játékot. A játékbéli bolt nagyon sok dolgot tartalmazhatna és sokféleképpen lehetne megoldani. Az egyik megoldás egy külön menü, ahol a felhasználó pénzből vásárolhat magának játékbéli pénzt. A játékbéli pénzt be lehetne váltani tartalmakra, amiktől élvezhetőbb lesz a játék. A karakterünk kaphatna különböző kiegészítőket, amik csak vizuális célt szolgálnának, például egy sapkát. A karakter textúrájára lennének ráhelyezve. Lehetne venni a karakterünknek másik textúrát is, vagy az ellenség textúráját megváltoztató tartalmat, ha a felhasználónak az alap textúra nem tetszik. Ezenkívül lehetséges részecske effektusokat is vásárolni, például, hogy a karakterünk futás közben mintákat húz maga után. A boltba belekerülhetnének olyan megvehető tartalmak is, amik a játékmenetet könnyebbé, gyorsabbá vagy izgalmasabbá teszik. Mint például a gyűjthető érmék értékére ráduplázó erősités vagy a halál utáni újra éledés az utolsó biztonságos pozícióban igy könnyebbé téve a játékmenetet.

# Összefoglalás

A Kocka Hősök program alatt sok mindent sikerült tanulnom. A játék készítése során sikerült megismernem a különböző menük elkészítését. A játékosoknak a beléptetését adatok küldését és olvasását szerverről. Ezen kívül sikerült jobban megismernem a unitynak a kezelő felületét.

A játékomon mind játék mind programozás szempontjából van mit csiszolni. Az elkészített osztályok átláthatóságán és a benne lévő kódnak is. De miközben írtam a programot is rengeteg újdonságot tanultam ezért vissza nézve a programomon rengeteg más dolgot is meg lehetett volna jobban megoldani.

A programnak a legnehezebb része volt az összeköttetést megcsinálni a szerver és unity között a php kódban az általam gyártott osztállyal. A saját osztályomnak a metódusait jól felhasználni, hogy könnyebben és átláthatóbban megtudjuk oldani a beléptetési rendszert miközben a beérkező hibákat is lekezeljük.

A program készítése közben szemelőt tartottam a könnyű pálya építést is ha később tovább akarom fejleszteni a játékot ezért extra figyelmet fektettem abba, hogy paraméterezhetőek legyenek a különböző pálya elemek. A pályák egyszerű elkészítése az általam leprogramozott objektumokból is külön fejtörést okozott. Ez tovább nehezítette a programnak az elkészítését.

De végeredményül elégedett vagyok a játékommal. Az akkori tudásom alapján szerintem megfelelően oldottam meg a feladatnak az elkészítését. Szerintem megfelelően működik a program. Látom magamon hogy rengeteg új tapasztalatot szereztem a programozás körében ezzel a projekttel.

# Felhasznált anyagok

## Fileok

* Betűtípus: <https://www.dafontfree.net/freefonts-truetypewriter-polyglott-f102970.html>
* Pálya: <https://assetstore.unity.com/packages/2d/environments/free-platform-game-assets-85838>
* Játékos, ellenfél, töltőképernyő: Pallagi Bence
* Játékoshoz és ellenfélhez felhasznált anyagok: <https://www.pngwing.com/en/free-png-zvwro> <https://www.pngitem.com/middle/iwTwi_transparent-angry-mouth-png-angry-cartoon-mouth-transparent/> <https://www.pngkit.com/bigpic/u2w7e6q8o0u2q8w7/>
* Portál: <https://giphy.com/stickers/ralph-rick-and-morty-3o7aD2d7hy9ktXNDP2>
* Intro hang: <https://freesound.org/people/original_sound/sounds/494979/>
* Háttérzene: <https://freesound.org/people/Mrthenoronha/sounds/522297/>

## Kód

A kódomhoz a PHPOszekotes osztályban a <https://docs.unity3d.com/ScriptReference/Networking.UnityWebRequest.Post.html> oldalon talált kódot használtam fel a különböző metódusok kialakításában.

A FelhasznaloPontok osztályban lévő metódusokat a <https://www.youtube.com/watch?v=og1zXHzl2cE> youtube videóban látott kód alapján írtam meg.

# Irodalomjegyzék

* <https://www.youtube.com/watch?v=og1zXHzl2cE&list=PLTm4FjoXO7nfn0jB0Ig6UbZU1pUHSLhRU&index=12> utolsó elérés dátuma: 2021.04.06.
* <https://www.udemy.com/course/unitycourse/> utolsó elérés dátuma: 2021.04.06.
* <https://www.w3schools.com/js/js_json_intro.asp> utolsó elérés dátuma: 2021.04.06.
* <https://docs.unity3d.com/Manual/index.html> utolsó elérés dátuma: 2021.04.06.
* <https://www.php.net/> utolsó elérés dátuma: 2021.04.06.
* <https://dev.mysql.com/doc/> utolsó elérés dátuma: 2021.04.06.
* <http://www.inf.u-szeged.hu/~vargalg/inProgress/Unity/> utolsó elérés dátuma: 2021.04.06

Melléklet