



**Eötvös Loránd Tudományegyetem**

**Informatikai Kar**

**Média- és Oktatásinformatika Tanszék**

# **Gaming Center – Játékterem a weben**

**Témavezető:**

Bernát Péter

egyetemi adjunktus ; informatikai  
tudományok doktora, matematika-  
informatika szakos tanár

**Szerző:**

Vida Bálint

Programtervező informatikus BSc.

Budapest, 2023

## Tartalomjegyzék

1	Bevezetés .....	4
1.1	Készítő és elérhetőségek.....	4
1.2	Szakdolgozati témabejelentő.....	4
1.3	A dolgozat felépítése.....	4
1.4	Motiváció .....	5
2	Felhasználói dokumentáció.....	6
2.1	Bevezetés .....	6
2.1.1	Rendszerkövetelmények .....	6
2.1.2	A program leírása .....	6
2.1	Oldaltérkép.....	7
2.2	Oldalak bemutatása .....	8
2.2.1	Kezdőlap .....	9
2.2.2	Játékok .....	10
2.2.3	Játékoldalak bemutatása .....	11
2.2.4	Kapcsolat .....	23
2.2.5	Profil .....	26
2.2.6	Bejelentkezési oldal.....	27
2.2.7	Regisztrációs oldal.....	29
3	Fejlesztői dokumentáció .....	32
3.1	Megvalósíthatóság .....	32
3.1.1	Erőforrás tervek .....	32
3.2	Követelmény leírás.....	32
3.2.1	Funkcionális követelmények, specifikáció .....	32
3.2.2	Felhasználói use-case diagramok.....	33
3.2.3	User-story diagram táblázat formájában.....	36
3.2.4	Bekért adatokra vonatkozó elvárások.....	41
3.2.5	Nem funkcionális követelmények, specifikáció .....	42
3.3	Megvalósítás .....	43
3.3.1	Használt technológiák .....	43
3.3.2	Drótvázterv.....	44
3.3.3	Modulok kapcsolatai .....	50
3.3.4	Üzleti logika .....	55
3.4	Futtatás .....	67

3.5 Tesztelés.....	71
4 Összefoglalás.....	78
4.1 Továbbfejlesztési lehetőségek .....	78
5 Források.....	78
5.1 Felhasznált szerkesztőprogramok.....	78
5.2 Felhasznált szakirodalom .....	79
5.3 Felhasznált dokumentáció .....	79
5.4 Egyéb.....	79

# 1 Bevezetés

## 1.1 Készítő és elérhetőségek

Név: Vida Bálint

Neptun kód: E93R1V

E-mail: [vbaint1997@gmail.com](mailto:vbaint1997@gmail.com)

E-mail: [e93r1v@inf.elte.hu](mailto:e93r1v@inf.elte.hu)

Szakdolgozat/Szoftver elkészítésének ideje: 2023. február 1. – 2022. május 29.

Weboldal (offline – Lokális szerver): Tömörítve csatolt állományok

Weboldal (online – Online): <https://github.com/S0me0ne97/GameCenter>

## 1.2 Szakdolgozati témabejelentő


A Gaming Center weboldal többféle játék kipróbálására biztosít lehetőséget. A játékok a weboldalon kategóriák szerint vannak rendezve, hogy mindenki könnyen meg tudja találni a saját ízlésének megfelelő játékot. Játék közben a felhasználónak lehetősége nyílik a játékokat szüneteltetni, elmenteni és betölteni. Ha a felhasználó korábban nem ismerte az adott játékot, akkor a játékdalon egy erre kialakított gomb megnyomásának a segítségével elolvashatja a játék leírását. A legjobb eredményeket elérő játékosok felkerülnek az adott játék ranglistájára a megszerzett pontjaik alapján. Amennyiben a felhasználó regisztrált és bejelentkezett, akkor a statisztikai elmentésre kerülnek, melyet a Profil fül alatt meg is tud tekinteni, mely fül csak a bejelentkezett játékosoknak elérhető. Regisztráción és bejelentkezésen kívül van lehetőség regisztráció nélkül játszani, továbbá hibabejelentést küldeni. A menüsorban lehetősége lesz a felhasználónak a Kezdőlap, Kategóriák, Regisztráció/Bejelentkezés, Kapcsolat és bejelentkezett felhasználó esetén Profil menüpontok között navigálni.

## 1.3 A dolgozat felépítése

A dokumentum több főbb részre van bontva.


A dolgozat **1. pontjában** található a bevezetés. Itt található a témabejelentő, rövid leírása a weboldalnak és a motiváció.

A **2. pont** tartalmazza a felhasználói dokumentációt. Kitérve az egyes típusú felhasználók jogosultságaira is, mint általános felhasználó, regisztrált felhasználó és admin. Ezeken kívül ez a rész tartalmazza az egyes aloldalak közti kapcsolatokat is.

A **3. pont** foglalja magába a fejlesztői dokumentációt.  Itt lesz megtalálható a szoftver felépítése, megvalósítása, architektúrája, az egyes játékok üzleti logikája, az adattárolásra szolgáló kód bemutatása, valamint egy tesztelési jegyzőkönyv.

## 1.4 Motiváció

Gyerekkoromban nagyon sok időt töltöttünk a testvéremmel különböző online játékokon. Volt olyan is, hogy egy egész délutánt bent töltöttünk a szobában a számítógép előtt ülve különböző online játékokkal játszva. Már akkor kíváncsian gondoltunk bele abba, hogy hogyan is kellene egy ilyen játékindo megcsinálni. Tervegettünk is sokat, hogy mi milyen játékokat csinálnánk, viszont ezt akkor mindenféle programozási tudás hiányában nem tudtuk megtenni. Akkor volt az a pillanat, amikor eldöntöttem, hogy egyszer már csak azért is létre fogok hozni egy ilyen weboldalt. Innen eredt az a tervem is, hogy programozó szeretnék lenni, amikor felnövök. Így most több mint 10 évvel később, amikor már sikerült megszereznem az ehhez szükséges tudást kézenfekvő volt, hogy megvalósítom gyerekkori vágyakozásaim tárgyát.

Az oldal célközönségének szánok minden olyan embert, aki osztozik az én online játékok iránti szenvedélyemen, vagy esetleg az olyan embereket, akik már kinőttek  ezt a szenvedélyüket, viszont szeretnék egy kicsit nosztalgiázni a gyerekkorukról. Az oldal stílusának kialakítása közben is ez a fajta nosztalgia lebegett a szemem előtt. Az oldal a 2000-es évek letisztultságára szeretne hasonlítani.

## 2 Felhasználói dokumentáció

### 2.1 Bevezetés

#### 2.1.1 Rendszerkövetelmények

- **Eszköz:** A weboldal eléréséhez szükséges egy személyi számítógép, notebook vagy laptop.
- **Kijelző felbontás:** A számítógépnek rendelkeznie kell egy legalább HD minőségű (1080 \* 1920) kijelzővel.
- **Operációs rendszer:** Legalább Windows 10 Home verzióval kell rendelkeznie a számítógépnek, lehetőség szerint Windows 11 Home vagy Windows 11 Pro verzió.
- **Böngésző:** Google Chrome, Microsoft Edge vagy Firefox.
- **Offline eléréshez** szükség van Visual Studio Code vagy egyéb olyan fejlesztői környezetre, amelynek van olyan beépített funkciója, amely lehetővé teszi a lokális szerveren való futtatást.

#### 2.1.2 A program leírása

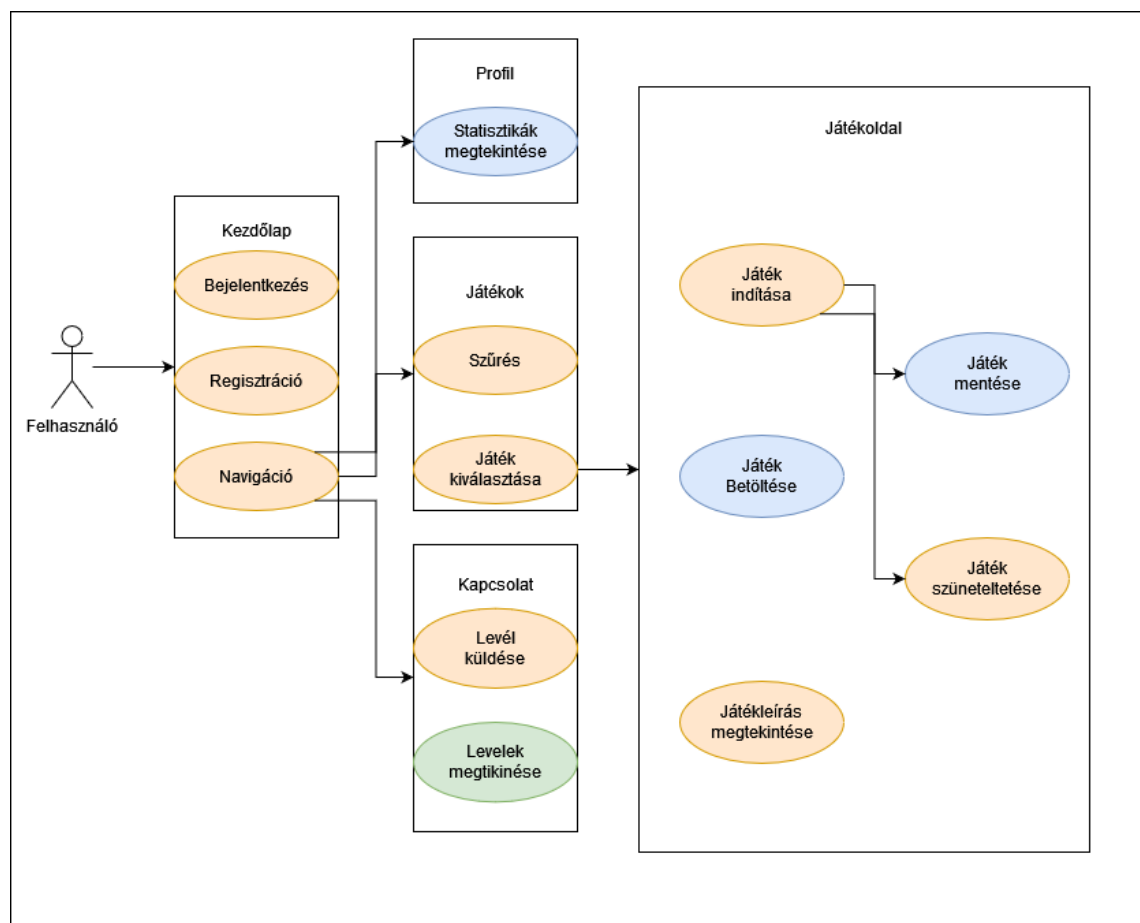
A weboldal célja egy olyan online felület létrehozása, amely lehetőséget biztosít játékok kipróbálására. A weboldal elérhetősége a dokumentum elején lett feltüntetve. A weboldal mindenki számára elérhető, azonban egyes funkciói a weboldalon való regisztrációhoz vannak kötve.

Az oldalhoz tartozó jogosultságok az alábbi táblázatban vannak feltüntetve:

	<b>Megnevezés</b>	<b>Megszerzésének módja</b>
1	Általános felhasználó	Minden felhasználó ezt a jogot kapja, amikor belép az oldalra
2	Regisztrált felhasználó	A weboldalon történő regisztráció során lehet ezt a jogosultságot megszerezni, amelyre minden Általános felhasználó jogosult
3	Admin	Ezt a jogosultságot csak akkor lehet megszerezni, amennyiben azt

		manuálisan beállítja egy olyan személy, aki hozzáfér a szerveren a tárolt felhasználók listájához.
--	--	--

## 2.1 Oldaltérkép



1. ábra Oldaltérkép

Az 1. ábrán látható a webhely oldaltérképe, ami bemutatja, hogy az egyes oldalakról milyen más aloldalakat és funkciókat ér el az adott felhasználó. Az index oldalt megnyitva a kezdőlapot találjuk magunk előtt.

Az egyes funkciókhoz tartozó színkódolás a jogosultsági szintekhez kapcsolódik. A barna színű funkciókhoz minden jogosultságú felhasználó hozzáfér. A kék színű funkciókhoz a regisztrált felhasználók férnek hozzá. A zöld színű funkcióhoz csak az admin felhasználók férnek hozzá. A regisztrált felhasználó hozzáfér az általános

felhasználó összes funkciójához. Az admin felhasználó hozzáfér a regisztrált felhasználó és az általános felhasználó összes funkciójához.



## 2. ábra általános menü

A 2. ábrán látható a főmenü rész, amely megtalálható minden oldalon, amelyet a felhasználó elér leszámítva a bejelentkezési és regisztrációs oldalt.

A főmenü részei:

- 3 Kezdőlap
- 4 Játékok
- 5 Kapcsolat
- 6 Profil – Ez az oldal csak bejelentkezett felhasználók számára jelenik meg a főmenü sávban
- 7 Bejelentkezés

## 2.2 Oldalak bemutatása

Ebben a pontban fogom részletezni az egyes oldalakon megtalálható funkciókat. A bemutatás során mindig a legkevesebb jogosultságú felhasználói szinttel (általános felhasználó) kezdem. Amennyiben az adott oldalon van olyan felhasználói szint (regisztrált felhasználó, admin), amelynek van jogosultsága más funkciókhoz is, akkor úgy értendő, hogy a magasabb jogosultsági szinttel rendelkező felhasználó mindig hozzáfér a kevesebb jogosultsággal rendelkező összes funkciójához, és még azon felül egy vagy több funkcióhoz. Ezek alapján nem fogok ismételtten leírni olyan funkciókat, amiket már azelőtt részleteztem az adott oldalon.

A menüsor és a lábjegyzet minden oldalon megtalálható, így annak a leírását csak egyszer a Kezdőlapnál részletezem.



## 2.2.1 Kezdőlap

### Menü



3. ábra általános menü, nincs bejelentkezett felhasználó

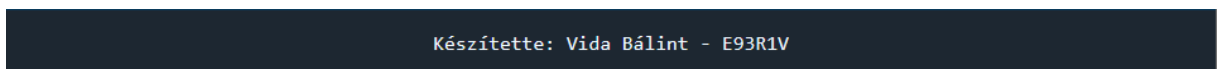


4. ábra általános menü, van bejelentkezett felhasználó

A lap tetején található a menü, amely 6 elemet tartalmaz. Ennek a segítségével lehet navigálni az egyes aloldalak között:

- A Gaming Center logója
- Kezdőlap
- Játékok
- Kapcsolat
- Profil
- Bejelentkezés / Kijelentkezés, amennyiben a felhasználó már be van jelentkezve

### Lábjegyzet

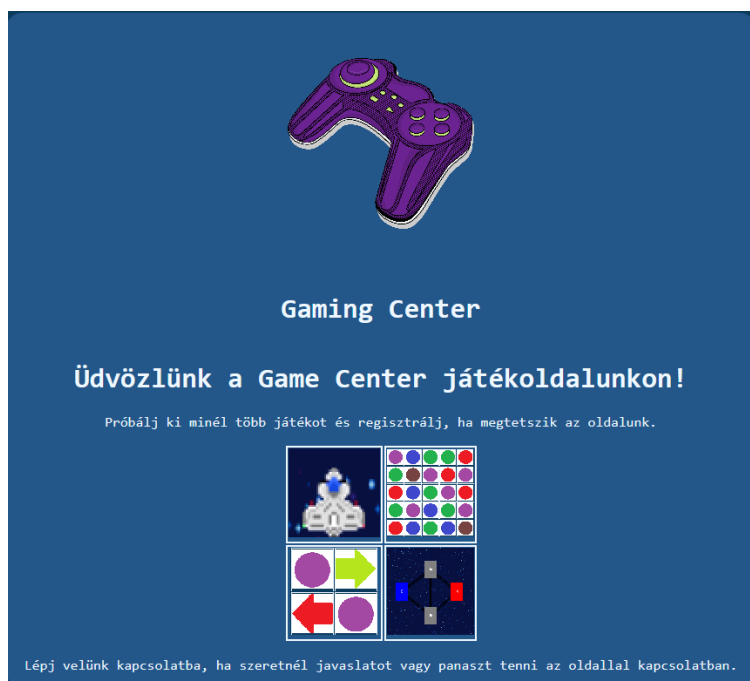


5. ábra lábjegyzet

Itt található egy rövid leírás arról, hogy ezt a weboldalt ki készítette.

### Üdvözlő szöveg

A kezdőoldalon kapott helyet egy üdvözlőszöveg és hangulatképek a játékokról.

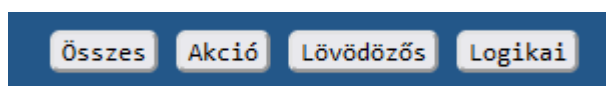


6. ábra kezdőlap

## 2.2.2 Játékok

### Kategóriák menüsor

Az alábbi képernyőképen látható a kategóriák menüsor.



7. ábra játék oldal szűrő

Ezeknek a gomboknak a megnyomása után szűrni lehet a játékokat típus szerint.

Amennyiben megnyomja a felhasználó valamelyik gombot, a menüsor alatt elhelyezkedő játékfelsorolás újratöltődik és így már csak az adott kategóriába vonatkozó játékok lesznek láthatóak.

Amennyiben a felhasználó ismét látni szeretné az összes játékot felsorolva, akkor az „Összes” gombot megnyomva újra betöltődik az eredeti sorrendben az összes játék.

### Játékok felsorolása

Az alábbi képen láthatóak a játékok felsorolva:



8. ábra játékok felsorolása

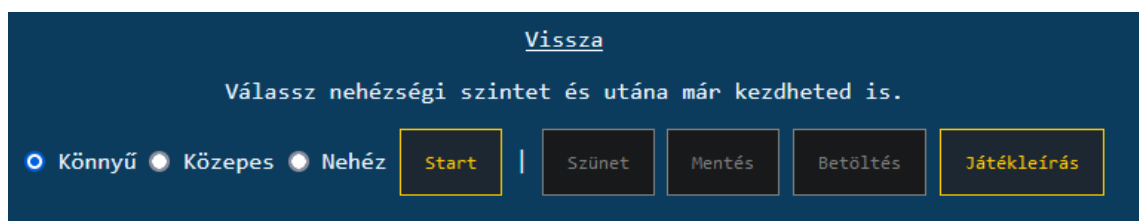
A játékok egymás alatt vannak felsorolva. Egy <sup>itt</sup> felsorolt játék tartalmazza a játék nevét, típusát, egy képet a játékról. Ha a felhasználó kiválasztott egy játékot, nincs más dolga, mint megnyomni a játékról készült képet, és be is töltődik a játék.

### 2.2.3 Játékoldalak bemutatása

A következőkben bemutatom az egyes játékok aloldalait, valamint az adott játékok felhasználói leírását.

#### 2.2.3.1 Space Shooter


##### Játék menüsáv



9. ábra játék menüsáv

A játék menüsávban találhatóak a játékot befolyásoló funkciók.


A menüsáv tetején található egy „Vissza” gomb, amelynek a segítségével a felhasználó vissza tud térni a játék kiválasztó oldalra. Ilyenkor a játék azonnal véget ér mentés nélkül.

A bal oldalon található  gombok segítségével lehet beállítani a játék nehézségét. Itt alapvetően a könnyű játékmód van kiválasztva, így, ha a játékos nem választ ki mást, úgy a „Start” gomb lenyomása után könnyű módban indul el a játék.


A nehézségi szint kiválasztása mellett található a „Start” gomb, amelynek a segítségével a játék nehézségi fokának a kiválasztása után el is tudja kezdeni a felhasználó a játékot.

A következő gomb a játék szüneteltetésére alkalmas „Szünet” gomb. Ha a játék még nincs elkezdve, akkor ez a gomb nincs még aktiválva, ezt úgy jelzi a felület, hogy ilyenkor a gomb szürke színű.

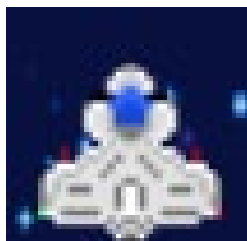
Mellette találhatóak a játék mentése és betöltése gombok. Általános felhasználóként ilyen funkciókra nincs jogosultság, emiatt ezek nincsenek aktiválva.

Végül a „Játékleírás” gomb, amit, ha megnyom a felhasználó, akkor megjelenik a játék menüsáv alatt a játékleírás. 

### Játékleírás

A játékleírás gomb megnyomására jelenik meg ez a felület. Itt részletesen van leírva, hogy mi a játék célja, milyen gombok lenyomásával lehet irányítani a játékot és végül a  pontszámítás módja is itt található meg.

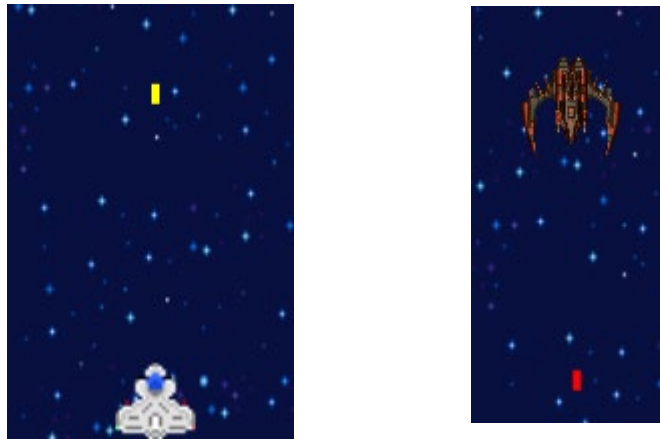
A Space shooter játék célja az, hogy minél tovább életben maradjunk az űrhajónkkal úgy, hogy közben minél több „ellenséges” űrhajót elpusztítunk. Ellenséges űrhajó mindegyik olyan egység, amit nem a felhasználó irányít.



10. ábra űrhajók, balra a játékosé, jobbra az ellenségé

Az ellenséges űrhajók a játéktér tetejéről indulnak véletlenszerű időközönként a játéktér alja irányába. A felhasználó által irányított űrhajó a játéktér alján

található meg, és a „Balra” és „Jobbra” nyilakkal lehet mozgatni, ami a billentyűzeten található meg. A játékos és az ellenséges űrhajó is képes lőni a másik irányába. A játékosnak az „X” gomb lenyomásával van erre lehetősége, míg az ellenséges űrhajók ezt random teszik meg.



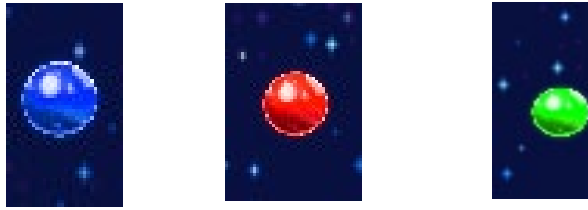
11. ábra lövedékek, balra a játékosé, jobbra az ellenségé

A játékos űrhajójából sárga színű lövedékek, míg az ellenséges űrhajókból piros színű lövedékek indulnak. A lövedékek amint elindultak függőlegesen haladnak a játékfelület teteje (amennyiben a játékos lőtte ki) vagy a játékfelület alja (amennyiben az ellenséges űrhajókból származik) irányába.

Amennyiben valamelyik, a játékos által indított lövedék eltalál egy ellenséges űrhajót úgy az adott ellenséges űrhajó elpusztul, ilyenkor, ahol a találatot kapott űrhajó „felrobban”. Ha a játékost találja el egy ellenség által kilőtt lövedék, úgy a játékos életerejé csökken egyel. Ha a játékos életerejé eléri a nullát úgy a játékos űrhajója felrobban és vége a játéknak. Amennyiben a játékos és egy ellenség összeütközik, úgy a játékos űrhajója megsemmisül eltekintve attól, hogy mennyi élete volt a játékosnak.

Van lehetőség erősítések felvételére is. Véletlenszerű időközönként jelennek meg erősítések a játékfelület tetején.

3 fajta erősítés jelenhet meg:



12. ábra erősítések

- Piros – extra élet, ha a játékos űrhajójával ütközik egy piros erősítés, akkor a játékos életeinek a száma <sup>egyel</sup> megnő
- Kék – extra mozgási sebesség, ha a játékos egy ilyen erősítést vesz fel, akkor bizonyos ideig megnő az űrhajó mozgási sebessége
- Zöld – halhatatlanság, ennek az erősítésnek a hatására egy bizonyos ideig nem vesz fel sebzést az űrhajó

A játéknak akkor van vége, ha a játékos összes élete elfogy.

Pontszámítás:

Minden, a játékos által elpusztított űrhajó után nő a játékos pontszáma tízzel.

### Játékfelület

A játékfelület a játék menüsáv alatt jelenik meg amint a játékos elindította a játékot a „Start” gomb megnyomásával.



13. ábra Space Shooter játékelület

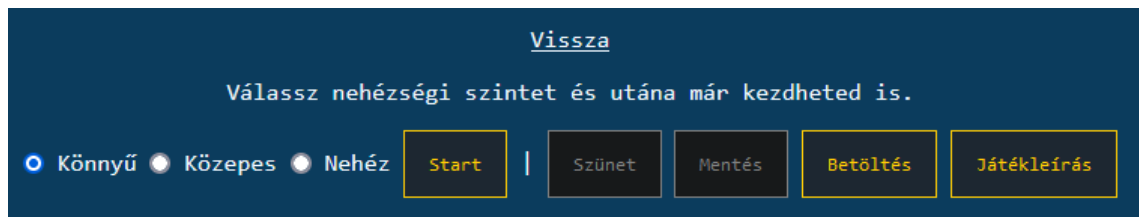
A bal felső sarokban találhatóak a pontok, az űrhajó még jelenlegi életereje és az egyes képességek hatásából hátramarad idő. Ha jelenleg az adott képesség nem aktív, akkor 0 szekundumot ír.

### Ranglista

A játékelület alatt helyezkedik el a ranglista, ahol megtekinthetőek a legjobb pontszámokat elért felhasználók.

### Regisztrált felhasználói jogosultságok

Amennyiben a felhasználó be van jelentkezve, úgy a játék menüsávban elérhetővé válnak számára a „Mentés” és „Betöltés” funkciók is.

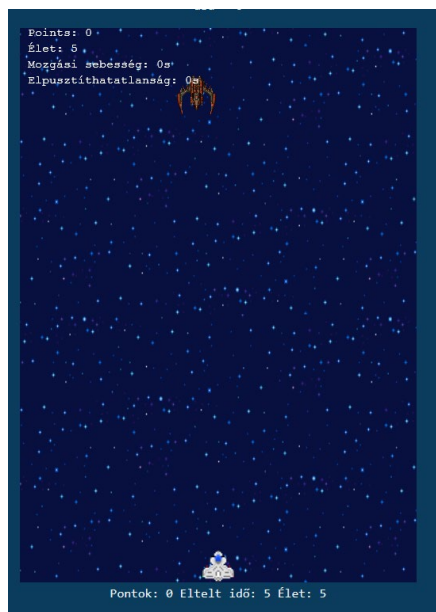


14. ábra játék menü, regisztrált felhasználó

A mentés funkció csak akkor lesz elérhető, ha egy játék már el lett kezdve.

Amennyiben a játékos egy aktív játékban van és megnyomja a „Mentés” gombot, úgy a játék pillanatnyi helyzete elmentésre kerül. Később ezt a Betöltési fül alatt meg is találja.

A játékosnak van lehetősége betölteni már korábban általa elmentett játékokat. Ezt úgy tudja megoldani, hogy a menüsávban kiválasztja a „Betöltés” gombot, amely gombnak a megnyomására eltűnik a játékfelület és megjelenik a betöltési felület.



15. ábra betöltési oldal

A Betöltési felületen egymás alatt felsorolva találhatóak meg a játékos által elmentett játékállások. egy betöltési elemhez tartozik egy kép a mentett játék játékállásáról és alatta részletezve, hogy a játékosnak hány pontja van az adott játékállásban, mennyi idő telt el és hány élete van.

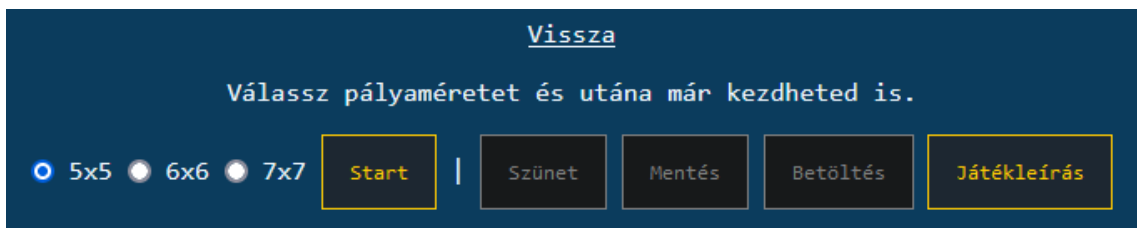


Amennyiben a játékos rákattint a játékállás képre úgy automatikusan betöltődik az a játék és folytatja a játékot a betöltött időpillanattól.

Amennyiben a játékos be van jelentkezve és lejátszott egy játékot, akkor az eredménye elmentésre kerül. Ha a játékos eredménye olyan magas, hogy be tud kerülni a top5 eredmények közé, akkor a ranglistán meg fog jelenni az eredménye a nevével.

### 2.2.3.2 Köss össze hármat

#### Játék menüsáv



16. ábra játék menüsáv

A játék menüsáv ennél a játéknál megegyezik a Space Shooter játéknál találhatóval, kivéve a nehézségi szint beállításnál nem nehézségi szint szerepel, hanem pálya méretek között lehet választani.

Pálya méretek:

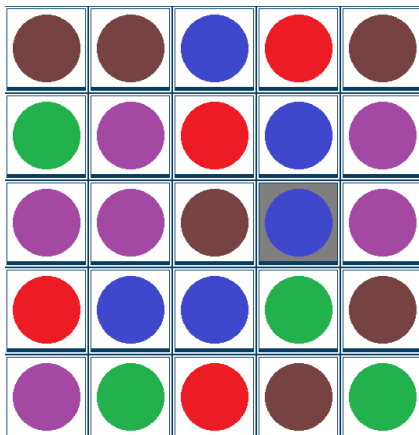
- 5 x 5
- 6 x 6
- 7 x 7

#### Játék leírás

A játék célja, hogy az 1 perc 40 másodperces játékidő alatt a lehető legtöbb pontot szerezzék meg a játékosok.

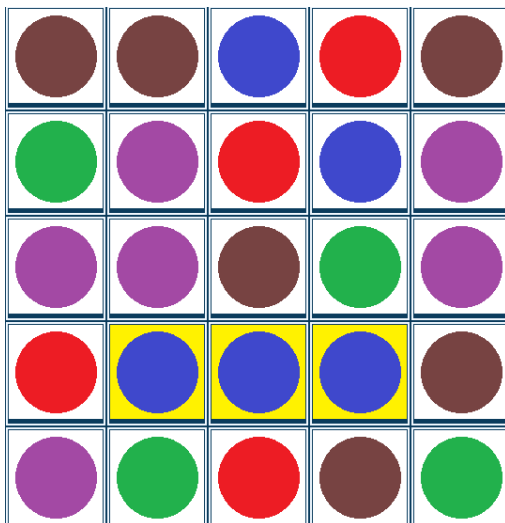
Pontokat úgy tud megszerezni a játékos, hogy két szomszédos mező kicserélésével 3, 4, vagy 5 azonos színű kört kell egy sorba vagy oszlopba rendeznie egymás mellé, és ekkor kap 50 pontot. Két szomszédos mezőt úgy tud kicserélni, hogy először kiválaszt egy mezőt kattintással. Ekkor a kiválasztott mező háttere szürkévé válik. Ezt vissza

tudja vonni úgy, hogy egy olyan mezőre kattint, aminek a kicserélésével nem keletkezik érvényes lépés.

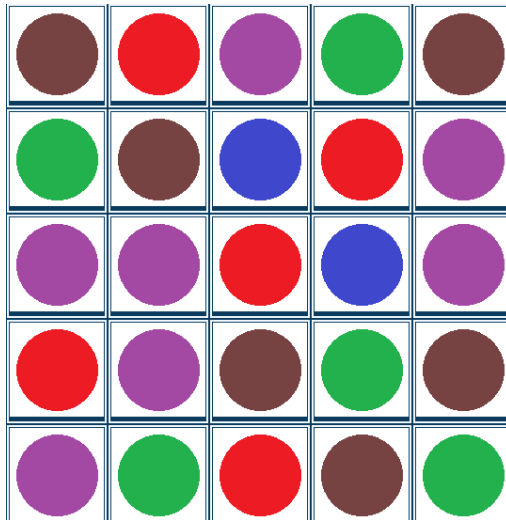


17. ábra első mező kiválasztva

Majd kiválaszt egy másik mezőt, amivel ki akarja cserélni. Ekkor, ha lehetséges ez a lépés, akkor a két mező kicserélődik, majd az így keletkezett 3 egy sorban vagy oszlopban egymás mellett elhelyezkedő mező háttere sárgává változik 1 másodpercig, majd eltűnnek.



18. ábra helyes lépés után



19. ábra játéktábla helyes lépés után

Ezután fentről „leesnek” a mezők. A pálya tetején pedig random mezőkkel fel lesznek töltve az üres mezők.

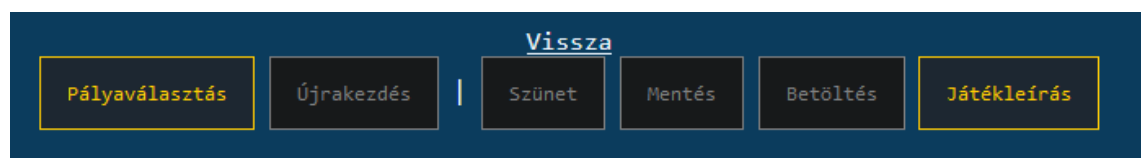
„Újratöltés” gomb arra szolgál, hogy ha a játékos nem talál olyan két mezőt, aminek a kicserélésével legalább 3 egyforma színű kör kerülne egy sorba vagy oszlopba egymás mellé, akkor 100 pont levonásával „vehet” egy újra generált pályát. Ebben az esetben a hátralévő idő és a pontok (100 pont levonása után) megegyeznek a játékos eddigi idejével és pontjaival.

## Regisztrált felhasználói jogosultságok

A regisztrált felhasználói jogosultságok megegyeznek a Space shooter játékkal.

### 2.2.3.3 Mini wars

#### Játék menüsáv



20. ábra játék menüsáv

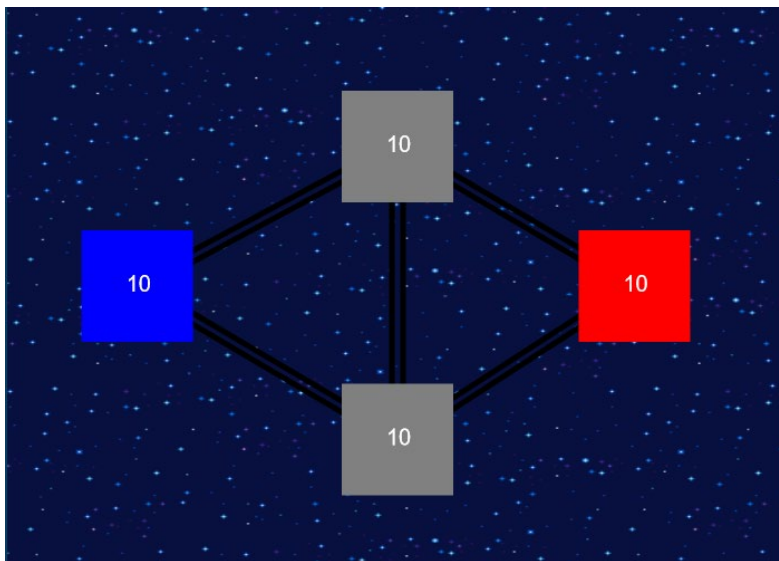
A játék menüsáv első eleme a „Pályaválasztás” gomb, amely megnyomásával megjelenik az összes lehetséges pálya, amik közül a felhasználó tud választani. Alapvetően az oldal betöltése után ez a képernyő jelenik meg.

A második elem az „Újrakezdés” gomb, ez eredetileg nincs aktiválva. Amennyiben a játékos elkezdett már egy játékot, akkor ennek a gombnak a megnyomásával tudja azt a pályát előlről kezdeni.

A menüsáv többi eleme megegyezik a többi játékkal.

### Játékleírás

A játékfelületen, ami az űrben helyezkedik el, különböző színű négyzetek és azokat összekötő utak vannak. A négyzetek bázisokat jelölnek és ha két bázis között van út, akkor ők szomszédosnak tekinthetők.



21. ábra játékfelület

A különböző színű bázisok:

- Kék – a játékos által birtokolt bázisok
- Piros – az ellenség által birtokolt bázisok
- Szürke – semleges bázisok

A bázisok közepén található számok jelzik, hogy az adott bázison mennyi egység található.

A pontszámítás úgy működik, hogy minden elfoglalt bázis után kap a játékos 50 pontot.

A játék célja, hogy a játékos elfoglalja az adott pályán található összes bázist. Ezt úgy tudja megtenni, hogy kiválaszt egy általa birtokolt mezőt, majd egy azzal szomszédos mezőt. Ekkor a játékos bázisán lévő összes egység megtámadja az elfoglalni kívánt bázist. Ha a játékos által küldött egységek száma nagyobb, mint a megtámadott bázison tartózkodó egységek száma, akkor a támadott mező el lesz foglalva és az ott tartózkodó egységek száma meg fog egyezni az eredetileg küldött egységek mínusz a védekező oldalon tartózkodó egységek számával.

Az ellenséges bázisok véletlenszerű időközönként véletlenszerű irányba fognak támadni. Abban az esetben, ha egy játékos vagy ellenség „megtámad” egy saját bázist, akkor nem fog csökkenni az egységek száma, hanem „erősítés” küldésnek fog számítani. Ekkor a fogadó bázison az egységek száma meg fog egyezni az eredetileg ott tartózkodó egységek számával plusz az érkező egységek számával.

„Egységtermelés” minden 3 másodpercben történik. Egységtermelés minden olyan bázison történik, amit ellenség vagy a játékos birtokol. Semleges mezőn nincsen egységtermelés.

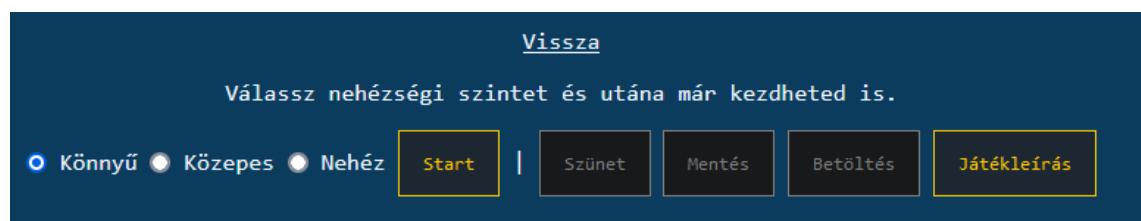
A pályának akkor van vége, ha vagy a játékos sikeresen elfoglalta az összes bázist, ami a pályán található, vagy az ellenség tette meg ugyanezt.

### Regisztrált felhasználói jogosultságok

A regisztrált felhasználói jogosultságok megegyeznek a Space shooter játékkal.

#### 2.2.3.4 Kártyás memóriajáték

##### Játék menüsáv



22. ábra játék menüsáv

A játék menüsáv megegyezik a Space shooter játékkal.

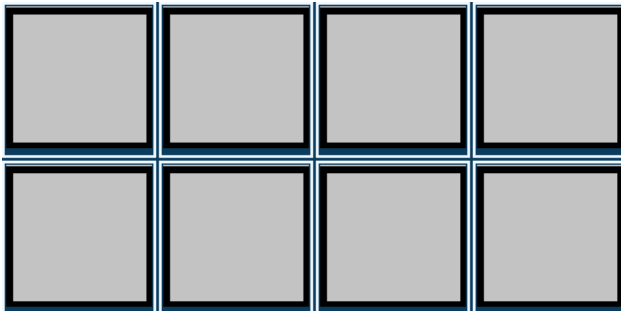
A nehézségi szintek ennél a játéknál arra szolgálnak, hogy különböző méretű pályákat generáljanak.

- Könnyű – 2 x 4 méretű pálya 4 különböző párral
- Közepes – 2 x 5 méretű pálya 5 különböző párral
- Nehéz – 2 x 6 méretű pálya 6 különböző párral

### Játékleírás

A játék célja, hogy a rendelkezésre álló 1 perc 40 másodperc alatt megtalálja a játékos az összes objektum párt.

A játék indításakor az összes mező lefordított állapotban van.

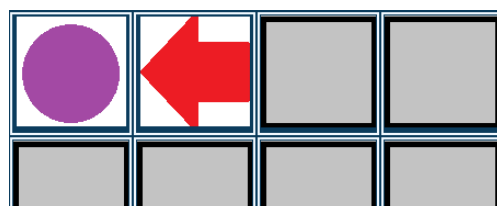


23. ábra játéktábla alap nézete

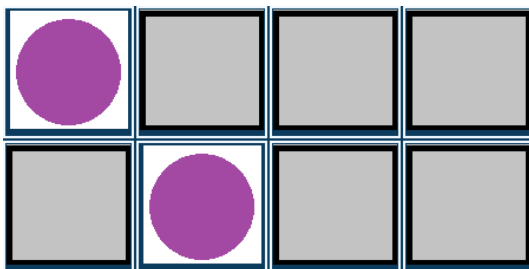
A játékos egy tetszőleges mezőre való kattintással tudja kiválasztani, hogy melyik mezőt szeretné megfordítani, azaz felfedni, hogy az adott mező milyen szimbólumot takar. Az első mező kiválasztása után a kiválasztott kártya felfordítva marad addig, amíg a játékos ki nem választ egy másik mezőt, amivel össze akarja hasonlítani az első mezőt. Ha a játékos kiválaszt egy újabb mezőt, akkor az is fel fogja fedni magát. Ha a két mező szimbóluma megegyezik, akkor a mezők felfordítva maradnak. Ha nem egyeznek meg a mezők, akkor 1 másodperc várakozás után a két kiválasztott mező visszafordul az eredeti állapotába.



25. ábra első mező kiválasztása



24. ábra második mező kiválasztása, nem egyezik a két mező



26. ábra második mező kiválasztása, egyezik a második mező

A játéknak akkor van vége, ha vagy lejár az idő, vagy a játékos megtalálja az összes párt.

A pontszámítás ennél a játéknál úgy működik, hogy minden megtalált pár után a játékos kap 50 pontot. A játék végén, ha a játékos megtalálta az összes párt, akkor a hátralévő idő alapján kap még extra pontokat (minden másodperc = 1 pont).

### Regisztrált felhasználói jogosultságok

A regisztrált felhasználói jogosultságok megegyeznek a Space shooter játékkal.

#### 2.2.4 Kapcsolat

A kapcsolat oldalon van a felhasználóknak panaszt vagy egyéb visszajelzést tenni az oldal adminjai felé.

# Lépjen velünk kapcsolatba!

Név:

E-mail cím:

Levél téma:

☒ Panasz  
☐ Visszajelzés

Szöveg:

Ide írja a szöveget...

27. ábra kapcsolat oldal

Amint az az előbbi képernyőképen is látszik a levél elküldéséhez meg kell adni a felhasználó nevét, e-mail címét, ki kell választani a levél témáját, amely alapértelmezetten Panasz, és meg kell adni a levél szövegét, amelyben a felhasználó kifejti a visszajelzését az oldallal kapcsolatban, vagy megadja a panasz tárgyát és annak az okát. Amint a felhasználó elkészült a levél szövegével, meg kell, hogy nyomja az „Elküld” gombot, amely kattintás hatására, amennyiben minden mező ki lett töltve, elküldi az adminok számára a levelet.

### Hibakezelés

Amennyiben a felhasználó megfelejtkezik valamelyik mező kitöltéséről, akkor ennek az elmulasztására felhívja a figyelmet egy szöveg, amely a ki nem töltött mező alatt jelenik meg, ahogyan az az alábbi képernyőfelvételen is látszik.



## Lépjen velünk kapcsolatba!

Név:   
Nem lehet üresen hagyni!

E-mail cím:   
Nem lehet üresen hagyni!

Levél téma:  
☒ Panasz  
☐ Visszajelzés

Szöveg:  
  
Nem lehet üresen hagyni!

28. ábra kapcsolat oldal, hibás kitöltés

### Admin jogosultság extra funkció

Az admin jogosultságú felhasználók számára az „Elküld” gomb alatt találhatóak meg a levelek, amiket a felhasználók küldtek számukra egymás alatt felsorolva egy vízszintes vonallal elválasztva egymástól.

Név: Gipsz Jakab  
Email: gipsz.jakab@pelda.hu  
Tárgy: complain  
Üzenet: Ide írja a szöveget...  
[Törlés](#)

29. ábra levelek megtekintése

Egy ilyen bejegyzés tartalmazza annak a felhasználónak nevét és e-mail címét, aki küldte a levelet. Továbbá a levél típusa és az üzenet szövege is látszik.

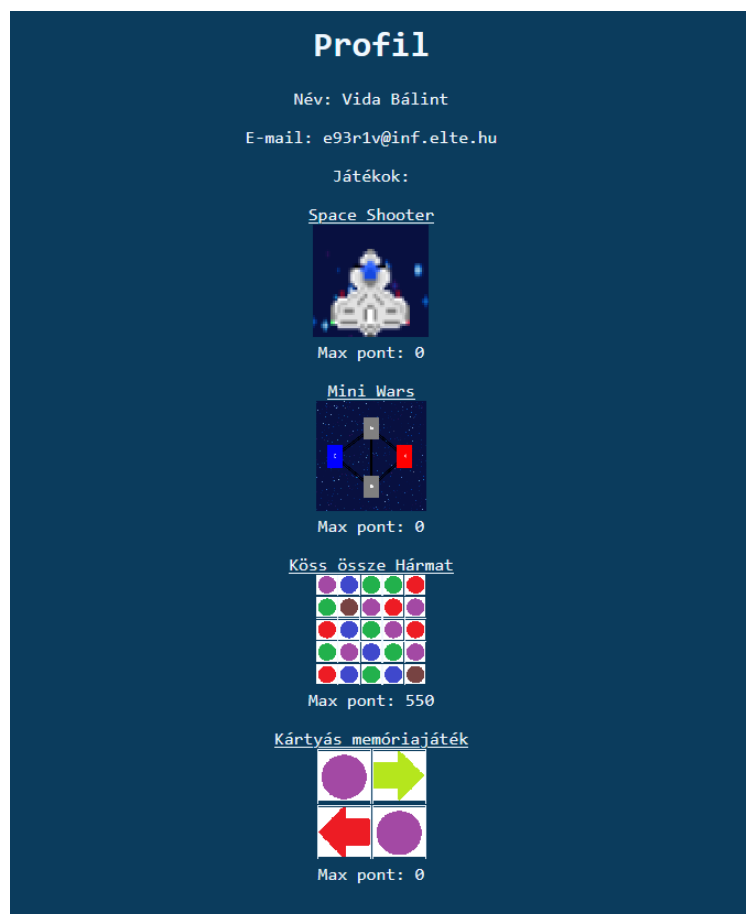
Az adminnak van lehetősége törölni a levelet, vagy a megkapott e-mail címre egy válaszlevelet küldeni (erre az oldalon nincs lehetőség, azt a saját e-mail fiókjából tudja megtenni).

### 2.2.5 Profil

A profil fül megtekintésére alap felhasználói jogosultsággal nincs lehetőség. Ezt az aloldalt csakis regisztrált felhasználók és az admin felhasználók tudják megtekinteni.

A megfelelő jogosultsággal rendelkező felhasználók itt tudják megtekinteni a saját, regisztráció során megadott, **adataikat**, továbbá játékokra lebontva a legjobb eredményeik is fel vannak sorolva.

Amennyiben a játék nevére, vagy a játék képére kattintanak át lesznek navigálva a kiválasztott játék oldalára.



30. ábra profil oldal

### 2.2.6 Bejelentkezési oldal

A felhasználónak, amennyiben korábban már regisztrált az oldalon úgy van lehetősége bejelentkezni, amelyet a felhasználói nevének és a jelszavának megadásával tud megtenni.



The image shows a login form on a dark blue background. At the top, the title "Bejelentkezés" is displayed in white. Below it, the label "Felhasználónév:" is followed by a white input field. Underneath, the label "Jelszó:" is followed by another white input field. A yellow button with the text "Bejelentkezés" is positioned below the password field. At the bottom, there are two links: "Vissza a főoldalra" and "Regisztráció", both underlined.

31. ábra bejelentkezési oldal

Amennyiben helyes adatok lettek megadva, úgy a „Bejelentkezés” gomb megnyomása után a felhasználó automatikusan vissza lesz navigálva a kezdőlapra. Bejelentkezett felhasználó esetén a főmenüsáv jobb oldalán a „Bejelentkezés” gomb szövege lecserélődik „Kijelentkezés” -re.

Amennyiben a felhasználó bejelentkezett úgy amíg ki nem jelentkezik, vagy be nem zárja a böngészőt addig be lesz jelentkezve ennek a weboldalnak az összes aloldalán. A felhasználónak a sikeres bejelentkezés után megnövekszik a jogosultsági köre általános felhasználóról regisztrált felhasználóvá.

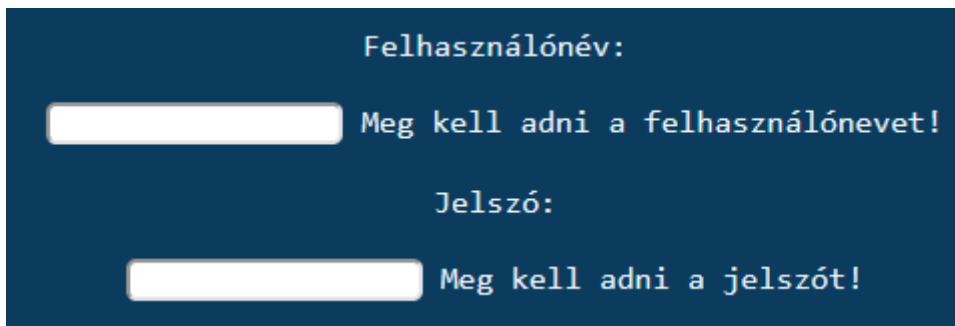
A „Kijelentkezés” gomb megnyomására a regisztrált felhasználó jogosultságai lecsökkennek általános felhasználóvá. A „Kijelentkezés” gomb szövege visszaáll „Bejelentkezés” -re.

Ha a felhasználó még nem rendelkezik regisztrált fiókkal, akkor van lehetősége új fiók regisztrálására a „Bejelentkezés” gomb alatt lévő „Regisztráció” gomb megnyomásával.

A bejelentkezési oldalról van lehetőség visszatérni a főoldalra.

### Hibakezelés

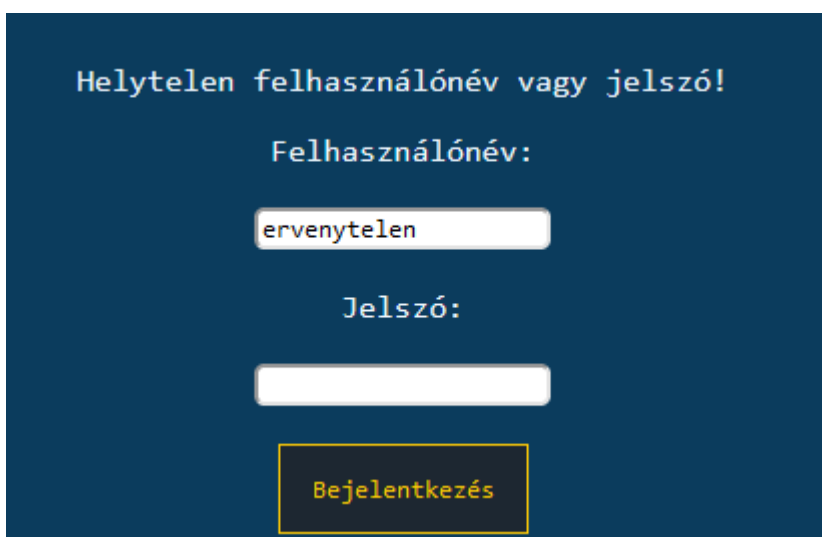
Amennyiben a megadott bejelentkezési adatok hiányoznak, akkor megjelenik az üresen hagyott mező (vagy mezők) mellett egy ennek az elmulasztására figyelmet felhívó szöveg.



A screenshot of a login form on a dark blue background. At the top, the text "Felhasználónév:" is displayed. Below it is an empty white input field, followed by the error message "Meg kell adni a felhasználónevet!". Further down, the text "Jelszó:" is shown. Below that is another empty white input field, followed by the error message "Meg kell adni a jelszót!".

32. ábra üresen hagyott mezők esetén a hibaüzenet

Amennyiben a megadott bejelentkezési adatok hibásak, akkor megjelenik a „Bejelentkezés” főcím alatt egy szöveg, amely felhívja erre figyelmet.




A screenshot of a login form on a dark blue background. At the top, the error message "Helytelen felhasználónév vagy jelszó!" is displayed. Below it, the text "Felhasználónév:" is shown. Underneath is a white input field containing the text "ervenytelen". Further down, the text "Jelszó:" is displayed. Below that is an empty white input field. At the bottom of the form is a yellow button with the text "Bejelentkezés".

33. ábra helytelen felhasználónév hibaüzenet

### 2.2.7 Regisztrációs oldal

A regisztrációs oldalon van lehetősége a még nem regisztrált felhasználóknak regisztrálni. Ehhez egy rövid űrlapot kell kitölteni, ahol megadja a leendő profilhoz tartozó adatait.



The image shows a registration form titled "Regisztráció" on a dark blue background. The form contains four input fields with labels in Hungarian: "Felhasználónév:" (Username), "E-mail:", "Jelszó:" (Password), and "Jelszó ismét:" (Repeat Password). Below the fields, there is a link "Vissza a főoldalra" (Back to the main page) and a yellow button labeled "Regisztráció" (Register).

34. ábra regisztrációs oldal

A regisztrációhoz szükséges adatok:

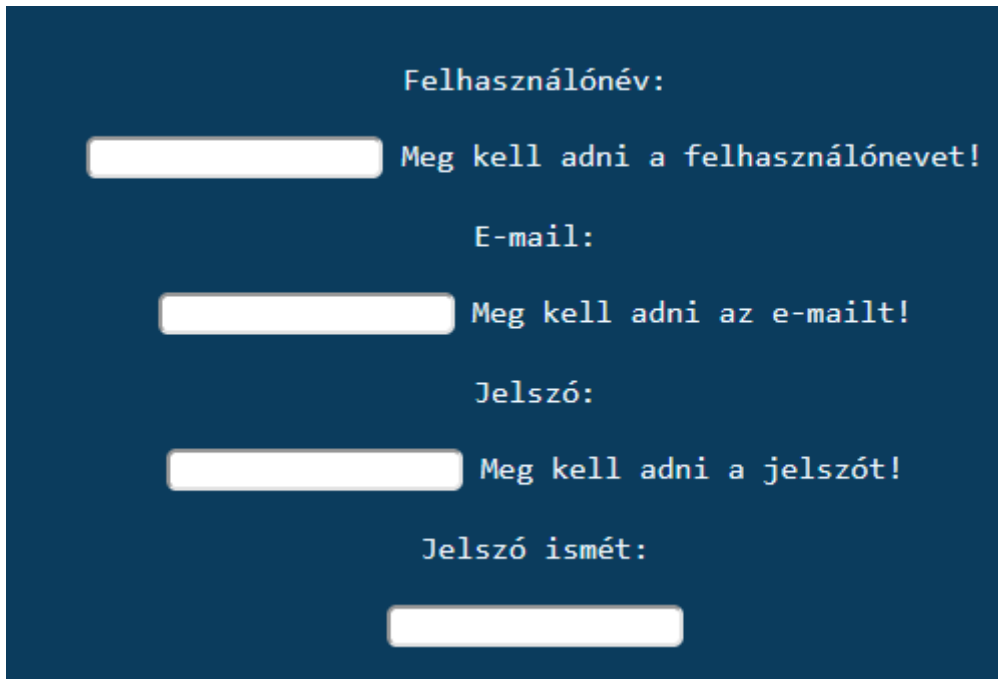
- Felhasználónév
- E-mail cím
- Jelszó
- Ismételt jelszó

Ha a felhasználó jól töltötte ki a regisztrációhoz szükséges adatokat, akkor a „Regisztráció” gomb megnyomása után vissza lesz navigálva a bejelentkezési oldalra, ahol most már be tud lépni a frissen regisztrált fiókjával.

Van lehetőség visszatérni a főoldalra

### Hibakezelés

Abban az esetben, ha a felhasználó üresen hagy egy vagy több kitöltendő mezőt, akkor a „Regisztráció” gomb megnyomása után nem történik meg a tényleges regisztráció. Ilyenkor az üresen hagyott egy vagy több mező mellett egy hibaüzenet jelenik meg, amely felhívja a felhasználó figyelmét arra, hogy nem töltötte ki az adott mezőt.

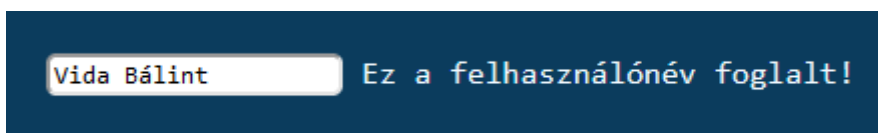


The image shows a registration form on a dark blue background. It contains four input fields, each with a red error message to its right. The labels and messages are as follows:

- Label: Felhasználónév: (User Name)  
Input field: [Empty]  
Error message: Meg kell adni a felhasználónevet! (You must provide a username!)
- Label: E-mail:  
Input field: [Empty]  
Error message: Meg kell adni az e-mailt! (You must provide an email!)
- Label: Jelszó: (Password)  
Input field: [Empty]  
Error message: Meg kell adni a jelszót! (You must provide a password!)
- Label: Jelszó ismét: (Repeat Password)  
Input field: [Empty]

35. ábra üresen hagyott mezők hibaüzenet

Ha a felhasználó által kiválasztott felhasználónév már foglalt, akkor a felhasználónév megadására szolgáló mező mellett egy hibaüzenet felhívja a figyelmet arra, hogy ez a felhasználónév már foglalt.

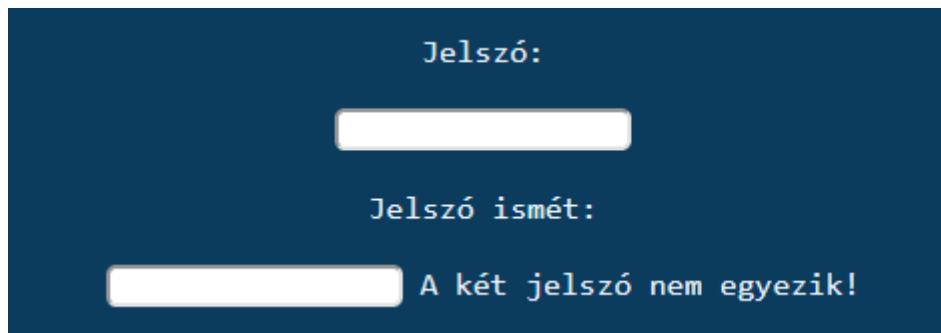


The image shows a registration form on a dark blue background. It contains one input field with a red error message to its right. The label and message are as follows:

- Label: [Empty]  
Input field: Vida Bálint  
Error message: Ez a felhasználónév foglalt! (This username is taken!)

36. ábra foglalt felhasználónév

Ha a két jelszó nem egyezik, akkor ezt is egy hibaüzenet fogja jelezni.



Jelszó:

Jelszó ismét:

 A két jelszó nem egyezik!

37. ábra két jelszó nem egyezik hibaüzenet

## 3 Fejlesztői dokumentáció

### 3.1 Megvalósíthatóság

#### 3.1.1 Erőforrás tervek

##### Szükséges erőforrások:

- **Humán erőforrások:** Tervezés: 50 óra

Fejlesztés: 300-350 óra

Tesztelés: 50 óra

Dokumentálás: 20 óra

- Hardver erőforrások: egy fejlesztői számítógép, 1 GB tárhely
- Szoftver erőforrások: fejlesztő környezet – Visual Studio Code, html5, css, javascript, php

adatbázis: json fileok

tesztkörnyezet: lokális php szerver

### 3.2 Követelmény leírás

#### 3.2.1 Funkcionális követelmények, specifikáció

A rendszer lényege: egy weboldal megvalósítása, ahol a felhasználónak van lehetősége játékokkal játszani akár bejelentkezés nélkül is, viszont bejelentkezés után kapjon extra funkciókat.

Ennek a megvalósításához a következő követelményeknek kell teljesülniük:

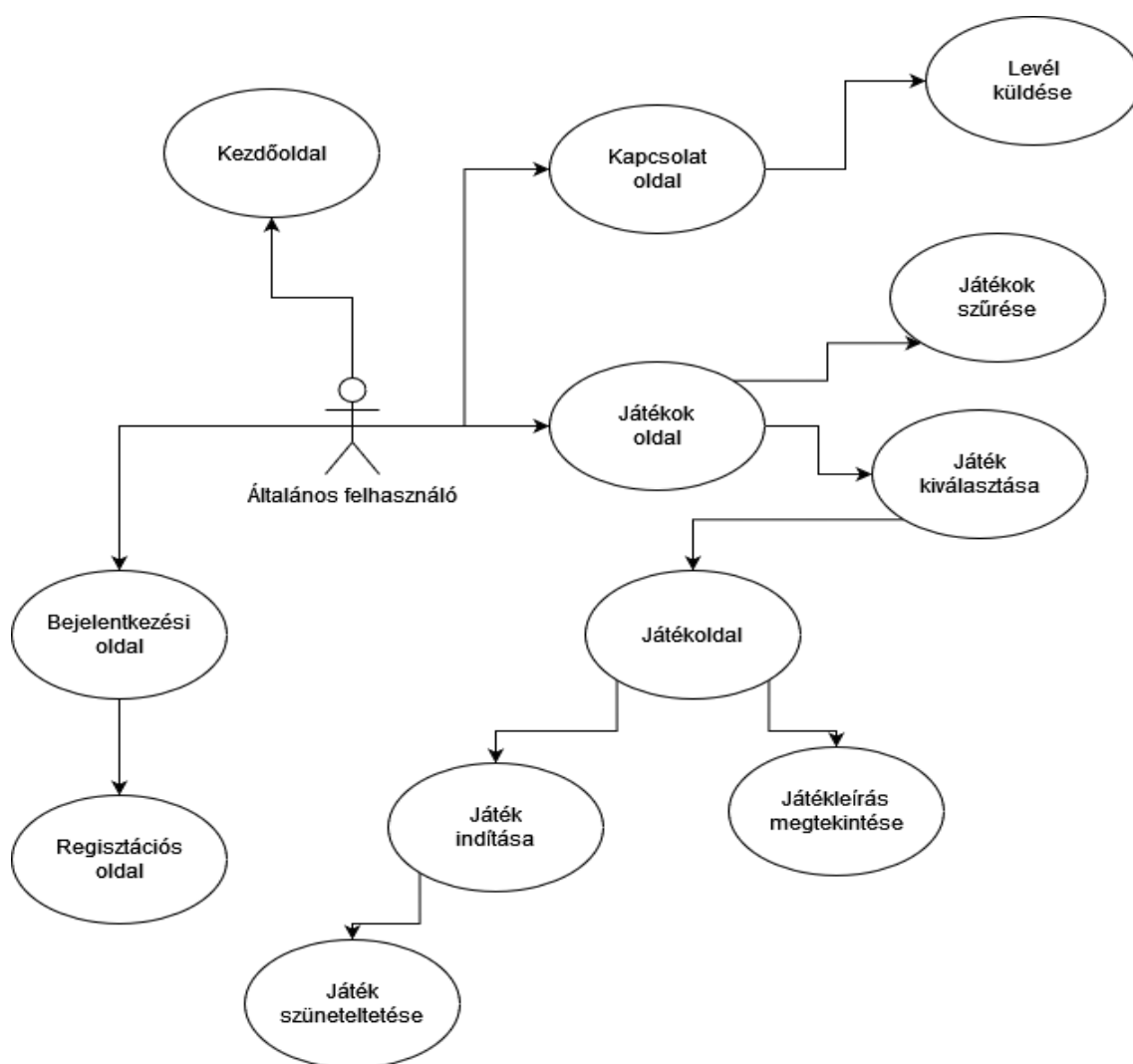
- A felhasználónak lehetőséget kell nyújtani játékok kipróbálására
- Lehetőséget kell nyújtani regisztrációra és bejelentkezésre
- A regisztrációt követően lehetőséget kell nyújtani extra funkciók eléréséhez, mint például játékok mentése betöltése.
- Bejelentkezett felhasználó esetén egy lejátszott játék végén el kell menteni az eredményét, és ha az alapján jogosult felkerülni az adott játék ranglistájára



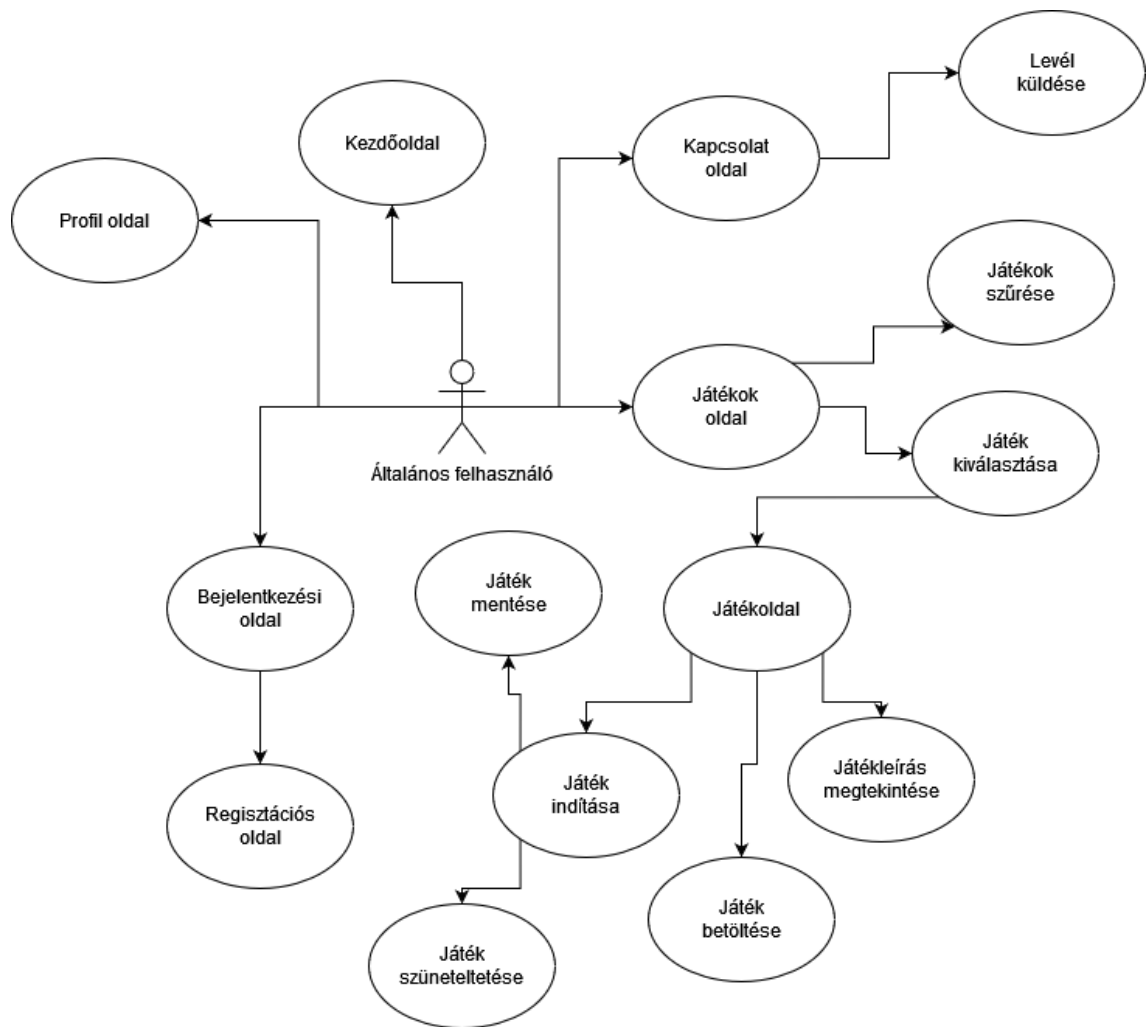
- Minden jogosultsági szintű felhasználónak van lehetősége a weboldalon belül panaszt vagy visszajelzést küldeni az adminok felé
- Az admin jogosultságú felhasználóknak van lehetőségük a visszajelzéseket és panaszokat megtekinteni
- A bejelentkezett felhasználóknak van lehetőségük megtekinteni a „Profil” oldalt, ahol nyomon követhetik az egyes játékokon belül elért legjobb eredményeiket.

### 3.2.2 Felhasználói use-case diagramok

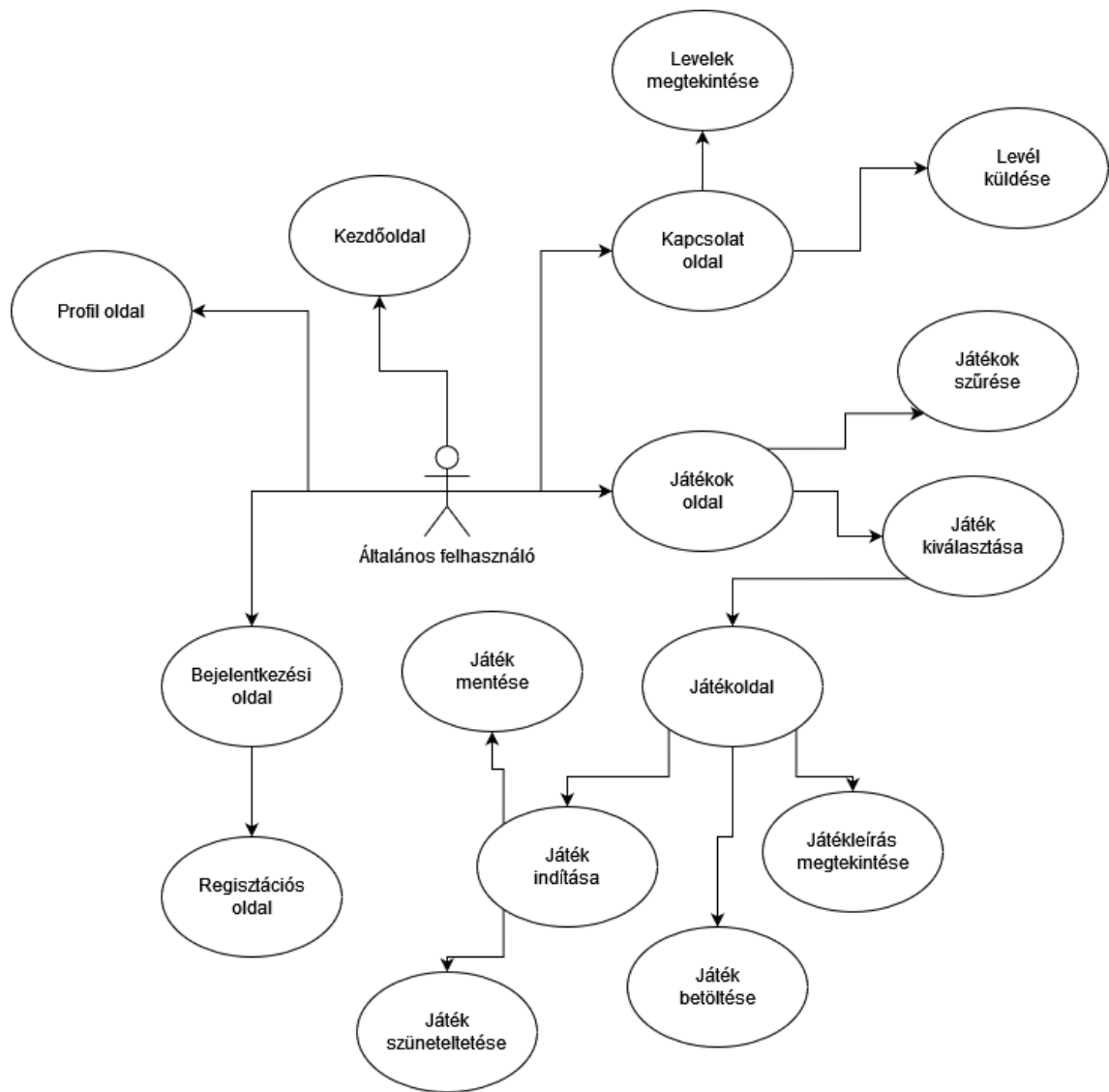
Különböző jogosultságú felhasználóknak különböző mértékű funkciókat érnek el, ennek az ábrázolásához 3 diagrammot használok:



38. ábra Általános felhasználó use-case diagram



39. ábra regisztrált felhasználó use-case diagram



40. ábra admin felhasználó use-case diagram

A jogosultsági körök tömören:

#### Általános felhasználó:

- A különböző aloldalak megtekintése
- Játékok kipróbálása
- Megtekintheti az egyes játékokhoz tartozó ranglistákat
- Játékleírás elolvasása
- Játékok szüneteltetése és annak a folytatása
- Panasz/visszajelzés küldése
- Lehetőség regisztrációra

### Regisztrált felhasználó:

- Minden, amihez az általános felhasználónak van joga
- Bejelentkezés
- Kijelentkezés
- Játékok mentése
- Játékok betöltése
- Az egyes játékokhoz tartozó ranglistákra van lehetősége felkerülni
- „Profil” aloldal megtekintése

### Admin:

- Minden, amihez a regisztrált felhasználónak van joga
- Panaszok/visszajelzések megtekintése

### 3.2.3 User-story diagram táblázat formájában

Mint	Adott	Amikor	Akkor
USER	Weboldal be van töltve	Navigációs fül egyik gombjára kattint	Betölt a megfelelő oldal
USER	Weboldal be van töltve	Bejelentkezés/Regisztráció gombra kattint	Betölt a bejelentkezés/regisztrációs oldal
USER	Bejelentkezés/regisztrációs oldal be van töltve	Helytelen adatokkal próbál belépni	Hibaüzenet megjelenik a kitöltendő szövegmezők alatt és nem történik bejelentkezés
USER	Bejelentkezés/regisztrációs oldal be van töltve	Helyes adatokkal próbál belépni	Bejelentkezés megtörténik, megjelenik a Profil fül a navigációs sávban
USER	Bejelentkezés/regisztrációs oldal be van töltve	Regisztrálni próbál helytelen adatokkal	Hibaüzenet megjelenik a kitöltendő szövegmezők alatt és nem történik regisztráció
USER	Bejelentkezés/regisztrációs oldal be van töltve	Regisztrálni próbál helyes adatokkal	Regisztráció megtörténik, bejelentkezés megtörténik, megjelenik a Profil fül a navigációs sávban

Mint	Adott	Amikor	Akkor
USER	Weboldal be van töltve	Megnyomja a kijelentkezés gombot	Kijelentkezés megtörténik, eltűnik a Profil fül a navigációs sávban
USER	Profil oldal be van töltve		Megjelenik egy táblázatban az összes játékhoz tartozó egyéni legjobb eredménye
USER	Kapcsolat oldal be van töltve	Üresen hagyja a kitöltendő mezőket	Hibaüzenet megjelenik a kitöltendő szövegmezők alatt és nem történik elküldés
USER	Kapcsolat oldal be van töltve	Kitölti a megfelelő mezőket helyes adatokkal majd próbálja elküldeni	Megtörténik az elküldés
USER	Játék kiválasztás oldal be van töltve	Kiválasztja a szűrési tulajdonságokat majd rányom a Szűrés gombra	A szűrésnek megfelelő játékok megjelennek
USER	Játék kiválasztás oldal be van töltve	Kiválaszt egy játékot kattintással	Betölt az adott játék a neki megfelelő oldalon
USER	Játékololdal be van töltve a megfelelő játékkal	Megnyomja a Játékleírás gombot	Megjelenik az adott játék leírása és a kezelési leírása
USER	Játékololdal be van töltve a megfelelő játékkal	Elkezd játszani	A játék elindul és megfelelően működik
USER	Játék oldal be van töltve a megfelelő játékkal, be van jelentkezve a felhasználó	Mentés gombra kattint	A játék elmentésre kerül a felhasználó neve alatt
USER	Játékololdal be van töltve a megfelelő játékkal, nincs bejelentkezve a felhasználó	Mentés gombra kattint	Nem történik semmi
USER	Játék oldal be van töltve a megfelelő játékkal, be van jelentkezve a felhasználó	Betöltés gombra kattint	Megjelennek a betölthető játékok, ha vannak ilyenek
USER	Játék oldal be van töltve a megfelelő játékkal, be van jelentkezve a felhasználó, megjelentek a betölthető játékok	Rákattint az egyik betölthető játékra	Betöltődik a játékállás és onnan folytathatja a játékot
USER	Játék oldal be van töltve a megfelelő játékkal, a játék folyamatban van	Rányom a játék szüneteltetése gombra	A játék állása szünetel, az idő mérése leáll
USER	Játék szünetel	Próbál tovább játszani	Nem történik semmi

Mint	Adott	Amikor	Akkor
USER	Játék szünetel	Megnyomja a játék folytatása gombot	Folytatódik a játék, az idő mérése folytatódik
USER	Játszik a felhasználó	Megnyomja az új játék gombot	Megkérdezi az oldal, hogy szeretné-e elmenteni az aktuális játékot

#### Space Shooter:

Adott	Amikor	Akkor
A játék indul.	A játékos megnyomja a "start" gombot.	A játék elkezdődik, és a játékos űrhajója megjelenik a képernyőn.
A játék folyamatban van	A játékos megnyomja az "lövés" gombot.	Egy lézer sugár indul az űrhajóból az ellenséges űrhajók felé.
A játék folyamatban van	A játékos a nyíl billentyűkkel mozgatja az űrhajót.	A játékos űrhajója a képernyőn kívánt irányba mozog.
A játék folyamatban van	A játékos felvesz egy erősítést.	Az űrhajó új képességet vagy erőt kap, például elpusztíthatatlan lesz.
A játék folyamatban van	Az űrhajó összeütközik egy ellenséges űrhajóval vagy akadállyal.	Az űrhajó életet, az ellenséges űrhajó vagy aszteroida pedig elpusztul.

Adott	Amikor	Akkor
Egy ellenséges űrhajó lő egyet		Egy lézer sugár indul az ellenséges űrhajóból a játékos űrhajója felé.

Mini wars:

Adott	Amikor	Akkor
A játék indul.		Az egységtermelés elindul minden játékos vagy ellenség által birtokolt mezőn.
A játék folyamatban van	A játékos kiválaszt először egy általa birtokolt mezőt, majd egy másik általa birtokolt mezőt	Az először kiválasztott mezőből az összes egység átkerül a másodjára kiválasztott mezőre.
A játék folyamatban van.	A játékos kiválaszt először egy általa kiválasztott mezőt, majd egy ellenséges vagy senki által nem birtokolt mezőt.	A játékos által birtokolt mezőről az összes egység támadni kezdi a másodjára kiválasztott mezőt.
A játékos megtámadott egy ellenséges vagy senki által nem birtokolt mezőt, a küldött csapatok száma kisebb, mint amennyien védik a mezőt.		A védekező mező nem lesz elfoglalva, annyival kevesebb védő marad, mint ahány egységgel a játékos támadott.
A játékos megtámadott egy ellenséges vagy senki által nem birtokolt mezőt, a küldött csapatok száma nagyobb, mint amennyien védik a mezőt.		A megtámadott mezőt már a játékos birtokolja.
Egy ellenséges mezőről támadás indul egy senki által nem birtokolt mezőre vagy a játékos által birtokolt mezőre, a küldött		A védekező mező nem lesz elfoglalva, annyival kevesebb védő marad, mint ahány egységgel az ellenfél támadott.

Adott	Amikor	Akkor
csapatok száma kisebb, mint amennyien védik a mezőt.		
Egy ellenséges mezőről támadás indul egy senki által nem birtokolt mezőre vagy a játékos által birtokolt mezőre, a küldött csapatok száma nagyobb, mint amennyien védik a mezőt.		A megtámadott mezőt már az ellenfél birtokolja.
A játékos vagy az ellenfél elfoglalta az összes mezőt.		A játéknak vége, a játékos megkapja a végső pontszámát.

Köss Össze Hármat:

Adott	Amikor	Akkor
A játék folyamatban van	A felhasználó két szomszédos kört kiválaszt, hogy felcserélje őket	Ha az áthelyezés egy három vagy sor, vagy oszlop kialakulásához vezet, akkor ezek a tárgyak eltűnnek, és a velük fentről követő körök leesnek. Új körök töltik fel a rács tetején lévő hézagokat.
A játék folyamatban van	A felhasználó kiválaszt egy kört	Annak a körnek a háttere szürke lesz, jelezve, hogy az a felhasználó kiválasztott köre.
A játék folyamatban van	Lejár az idő	A játék véget ér, és a végső pontszámot megjelenítik.
A játék folyamatban van	A felhasználó kiválaszt egy kört, majd egy másik kört, amelyet nem lehet az elsővel felcserélni	Semmi sem történik.
A játék folyamatban van	A felhasználó kicserél két kört, melynek következtében 3, 4 vagy 5 kör lesz egymás mellett egy sorban vagy oszlopban.	Eltűnik mindegyik kör, majd fel lesz töltve újra



### Kártyás memória játék:

Adott	Amikor	Akkor
A játék indul.	A játékos megnyomja a "start" gombot.	A játék elkezdődik, és a kártyák le vannak fordítva.
A kártyák le vannak fordítva.	A játékos kattint egy kártyára.	A kártya megfordul, és az ikonja látható lesz.
Egy kártya látható az ikonjával.	A játékos kattint egy másik kártyára.	Ha az ikonok megegyeznek, a két kártya eltűnik, és a játékos pontot kap. Ha nem, akkor mindkét kártya visszafordul.
A játék folyamatban van	A játékos megnyomja a "új játék" gombot.	Az összes kártya visszafordul, és a játék újra kezdődik.
A játék folyamatban van	A játékos megnyomja a "szünet" gombot.	A játék szünetel, visszatérhet a játékhoz.
A játék idő lejár	A játékosnak nem sikerült megtalálnia az összes párt	A játéknak vége.
A játékost megtalálta az összes párt		Megáll az időzítő, a játékos a maradék idő alapján kap extra pontokat.

A felsoroltak szükségesek az elvárt működéshez.

### 3.2.4 Bekért adatokra vonatkozó elvárások

#### Regisztráció

- Felhasználó név
- E-mail cím
- Jelszó kétszer

#### Bejelentkezés

- Felhasználó név
- Jelszó

#### Panasz/Visszajelzés küldése

- Felhasználó név

- E-mail cím
- Levél témája
- Levél szövege

### 3.2.5 Nem funkcionális követelmények, specifikáció

#### **Termék követelmények:**

##### Hatékonyság:

- Alacsony terhelés
- Gyors válaszidő

##### Megbízhatóság:

- Szabályos használat esetén nem fordulnak elő hibák
- Hibás bevitelnek le vannak kezelve és így sem fordul elő hiba

##### Biztonság:

- Adatvédelem

##### Hordozhatóság:

- Kompatibilis alacsony teljesítményű számítógépekkel is
- Kompatibilis a legtöbb böngészővel

##### Felhasználhatóság:

- Könnyen megérthető felület, kellő instrukciókkal ellátva
- Beszédes nevű funkciók, ami alapján mindenki tud asszociálni a funkciókra, akár külön instrukciók nélkül is

#### **Menedzselési követelmények:**

##### Környezeti:

- Nincsenek harmadik fél szoftver irányába való követelmények.

##### Működési:

- Hosszú és rövid távon is hatékonyan működik

Fejlesztési:

- Nincs külső könyvtár felé való függőség
- JavaScript, PHP, HTML5, CSS stíluskörnyezet
- Visual Studio Code környezet
- Firefox, Google Chrome, Microsoft Edge tesztkörnyezet
- Procedurális program

**Külső környezet:**

- Felhasználó adatait biztonságosan tárolja,
- Jogszabályoknak megfelel

### 3.3 Megvalósítás

A szoftver elkészítése 3 rétegű architektúrában készült:

- felhasználói felület
- üzleti logika
- adatbázis

A felhasználói felület kapcsolatban van az üzleti logikával és az adatbázissal is. Viszont az adatbázisból csak lekérni tudja az adatokat.

Az üzleti logika kapcsolatban van a felhasználói felülettel és az adatbázissal is. Le tudja kérni és módosítani is tudja mindkettő adatait. Nem épül osztályokra, procedurális a kialakítása.

Az adatbázis tárolja tartósan a játékokkal és a felhasználókkal kapcsolatos adatokat is.

#### 3.3.1 Használt technológiák

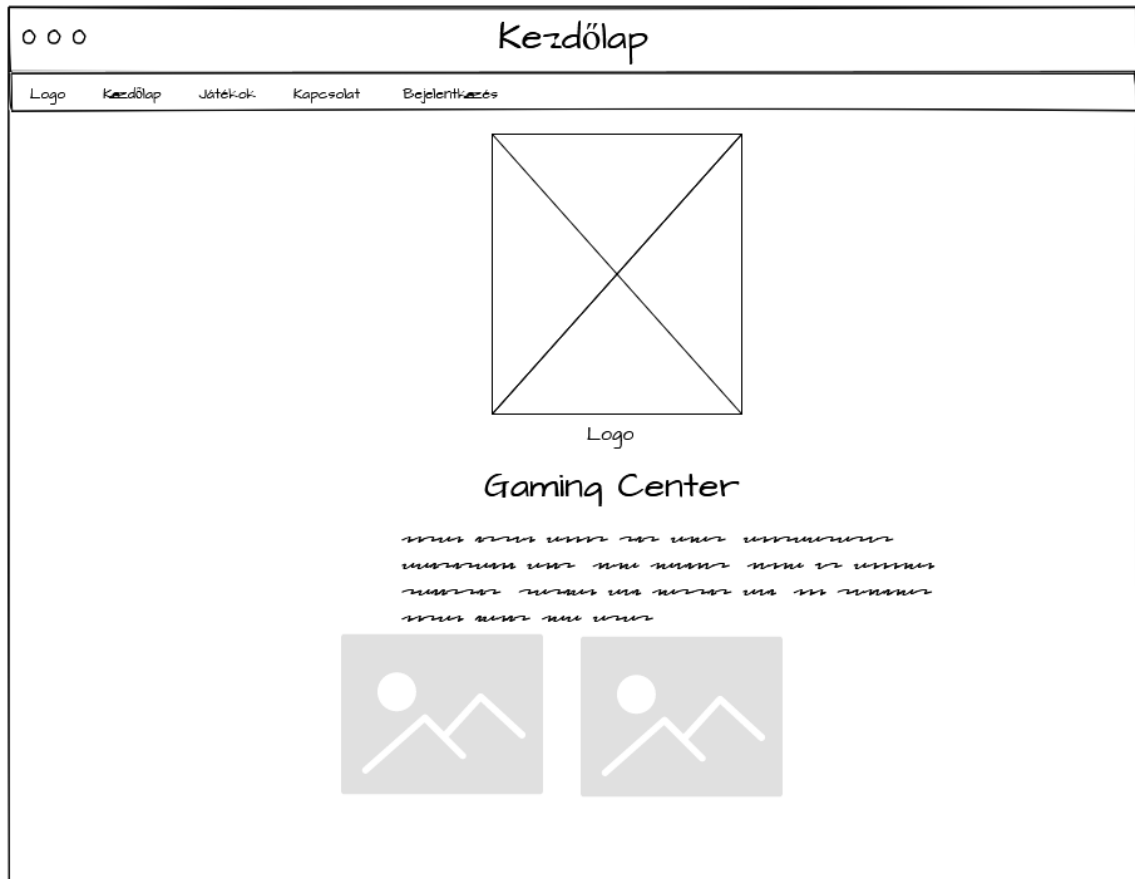
A felhasznált technológiákat az egyetemen való tanulmányaim során megismertek közül válogattam ki, ezek e következők:

- HTML5 – Hypertext Markup Language, a weboldalon megjelenő nézetet állítja elő.

- CSS – Cascading Style Sheets a dokumentumok megjelenési struktúráját állítja elő.
- PHP – Általános szerveroldali script nyelv.
- JavaScript – Objektorientált, prototípus-alapú szkriptnyelv.

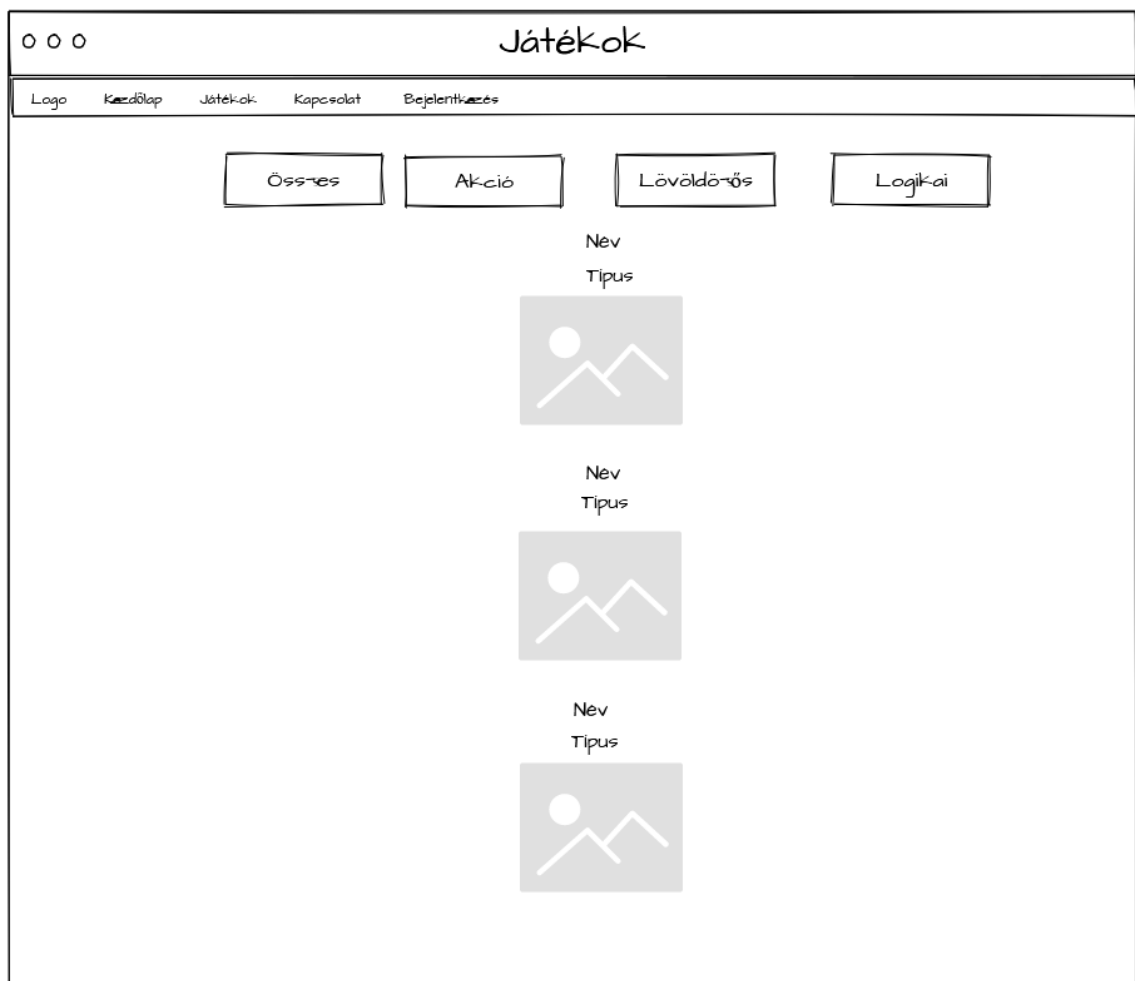
### 3.3.2 Drótvázterv

Kezdőlap:



41. ábra kezdőlap drótvázterv

Játékok:



42. ábra Játék kiválasztási oldal drótvázterv

Kapcsolat:

000

Kapcsolat

LogoKezdőlapJátékokKapcsolatBejelentkezés

Lépjön velünk  
Kapcsolatba

Név:

input

E-mail cím:

input

Levél téma:

☒ Panasz

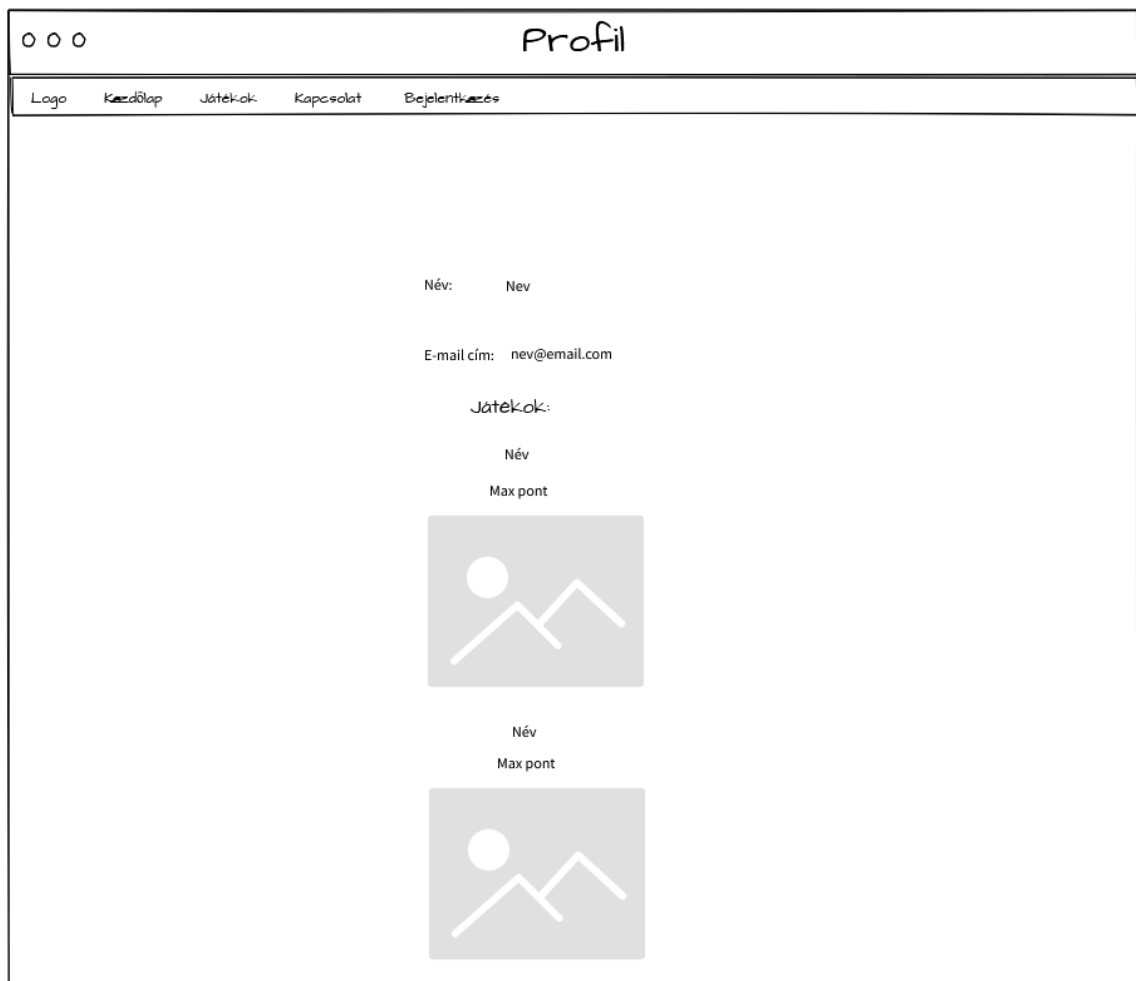
☐ Visszajelzés

Szöveg:

Küldés

43. ábra Kapcsolat oldal drótvázterv

Profil:



44. ábra profil oldal drótvázterv

Bejelentkezés:

ooo

Bejelentkezés

Bejelentkezés

Név

Jelző

Bejelentkezés

Regisztráció

45. ábra bejelentkezés oldal drótvázterv

Regisztráció:



ooo

Regisztráció

Regisztráció

Név

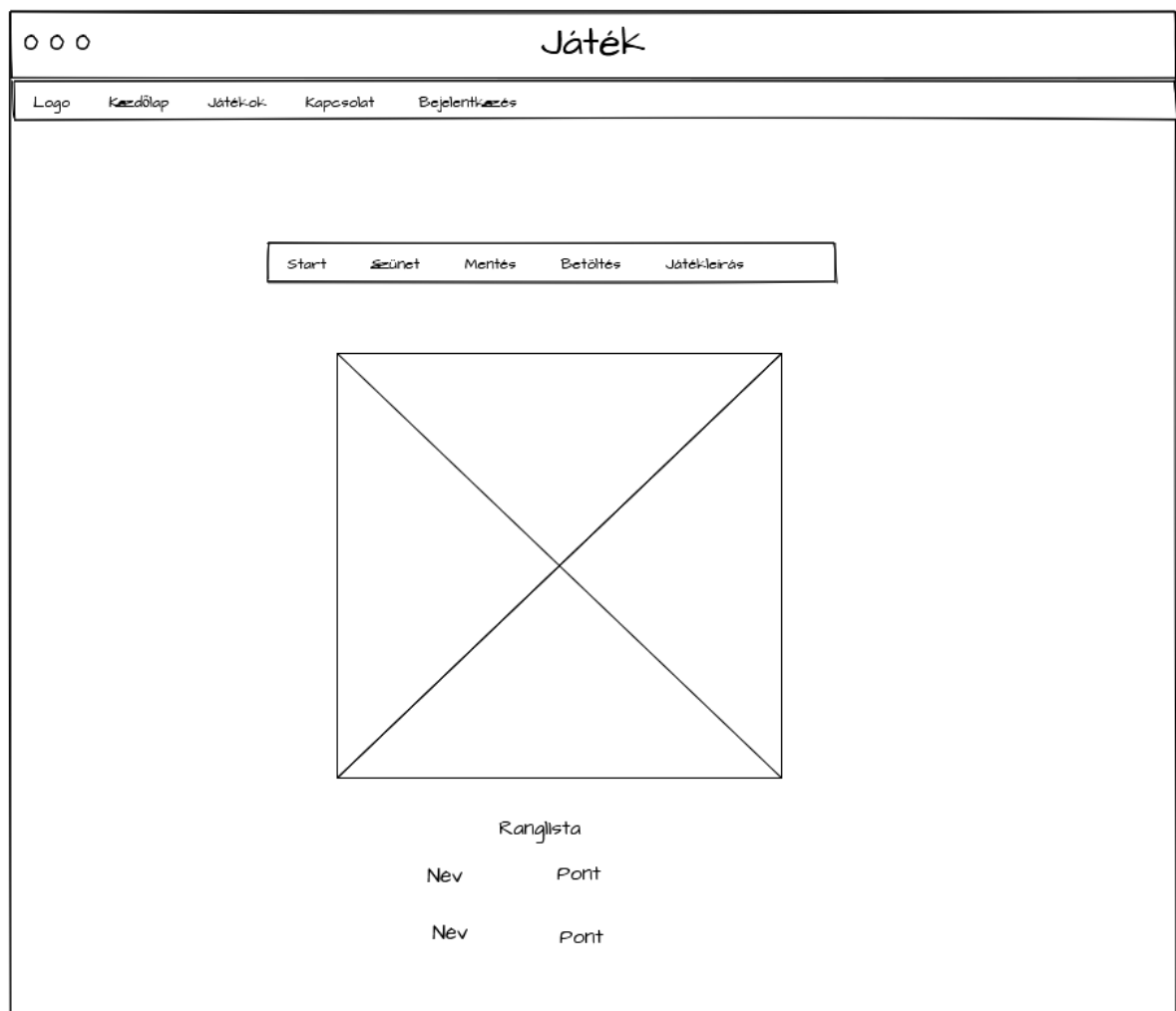
E-mail cím

Jelszó

Jelszó megegyezik

Regisztráció

Játékkoldalak:

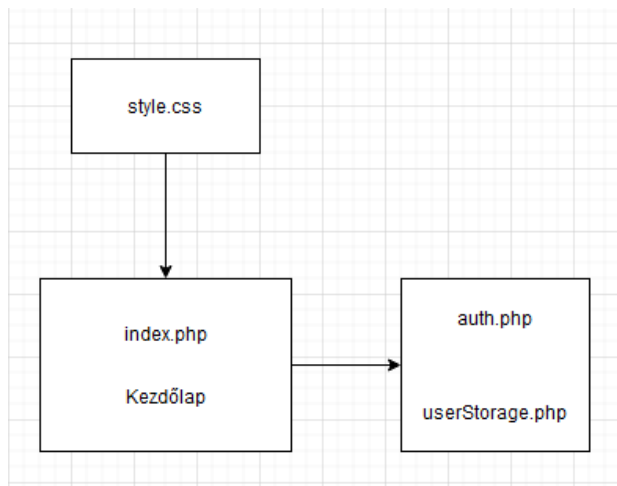


46. ábra Játék drótvázterv

### 3.3.3 Modulok kapcsolatai

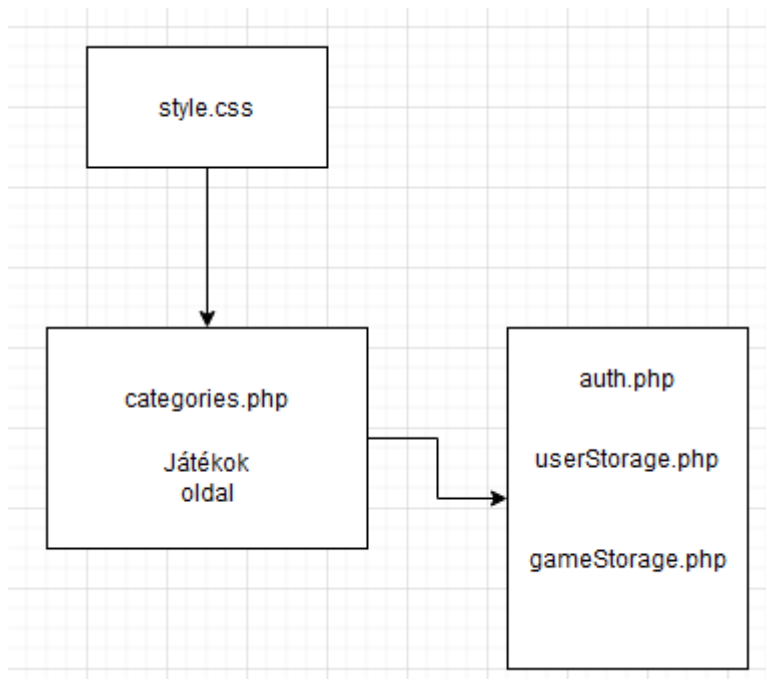
Ebben a pontban mutatom be azokat a modul kapcsolatokat, amelyek az egyes oldalaknál találhatóak.

Kezdőlap



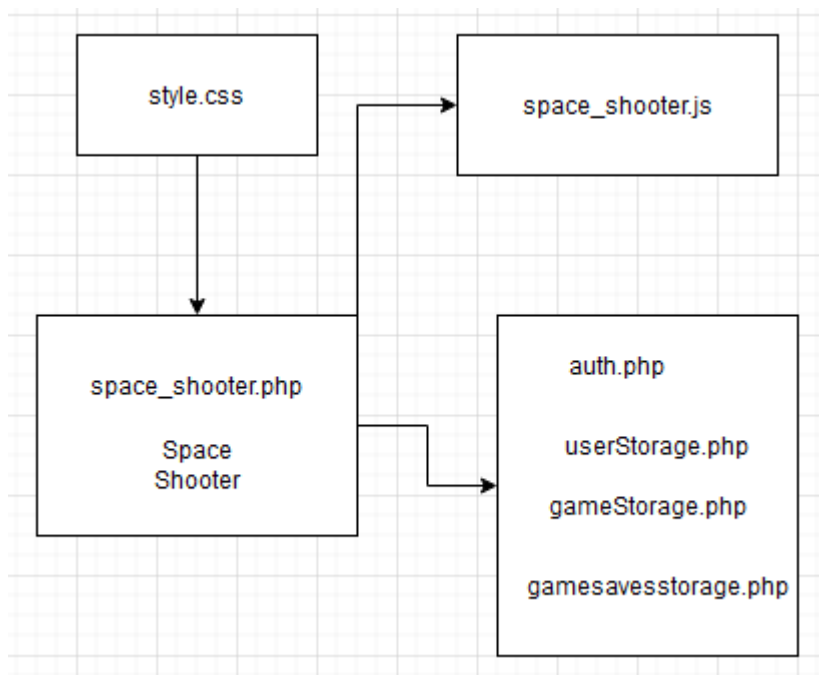
47. ábra kezdőlap modulkapcsolatok

## Játékok



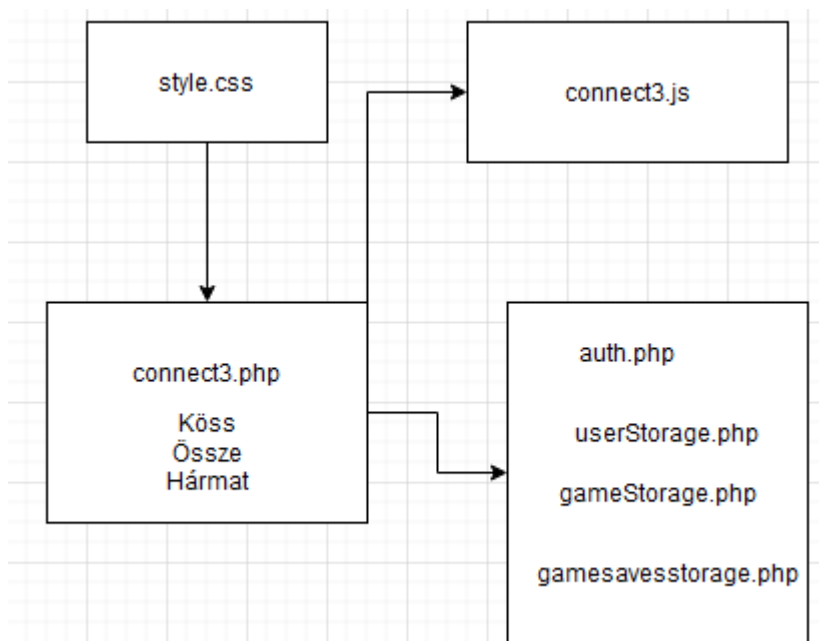
48. ábra játékok oldal modulkapcsolatok

## Space shooter



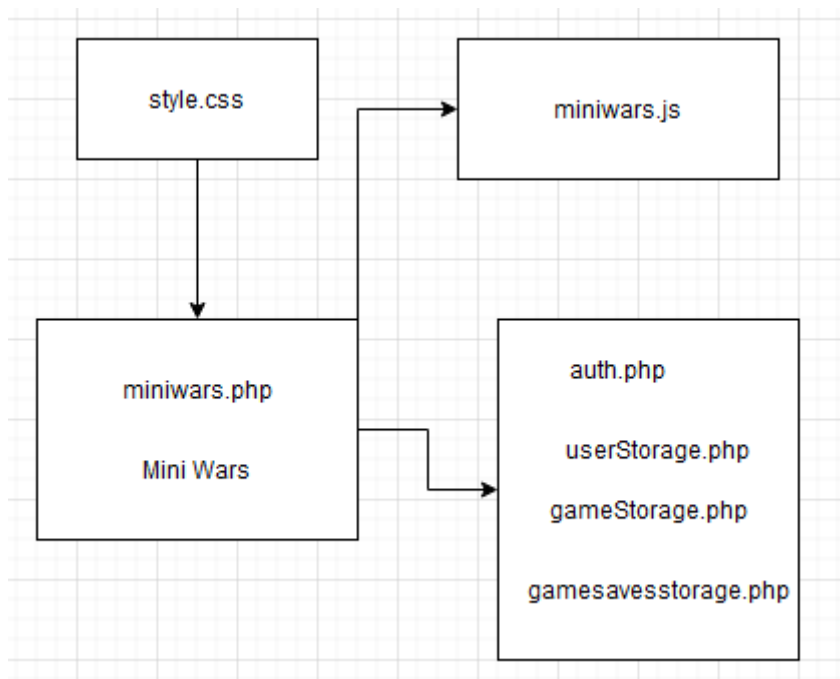
49. ábra Space shooter modulkapcsolatok

Köss össze hármát



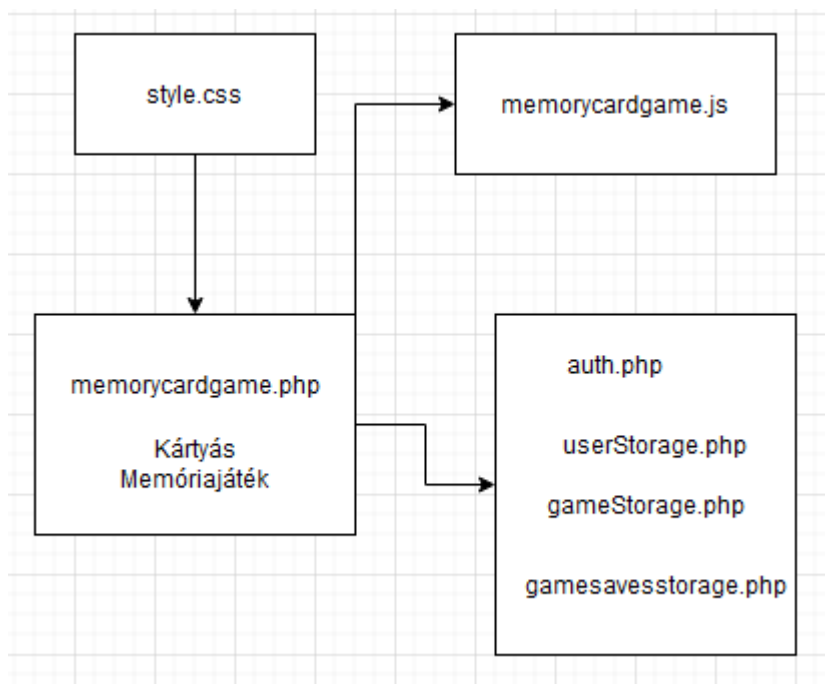
50. ábra Köss Össze Hármát modulkapcsolatok

Mini wars



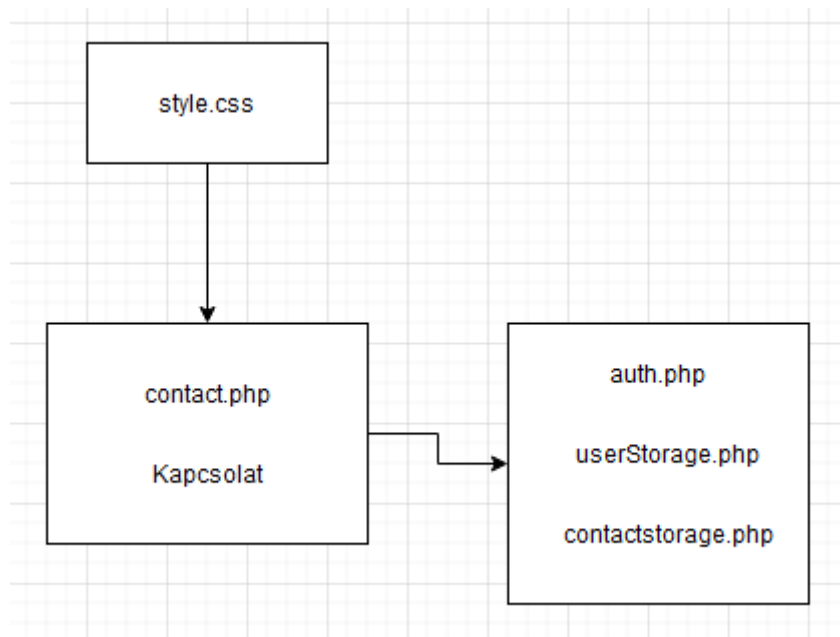
51. ábra Mini Wars modulkapcsolatok

## Kártyás memórijáték



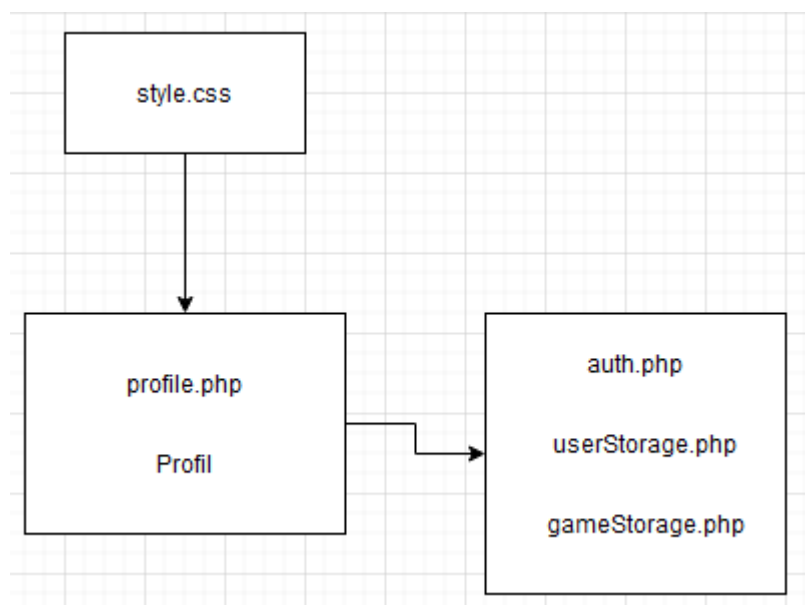
52. ábra Kártyás memórijáték modulkapcsolatok

## Kapcsolat



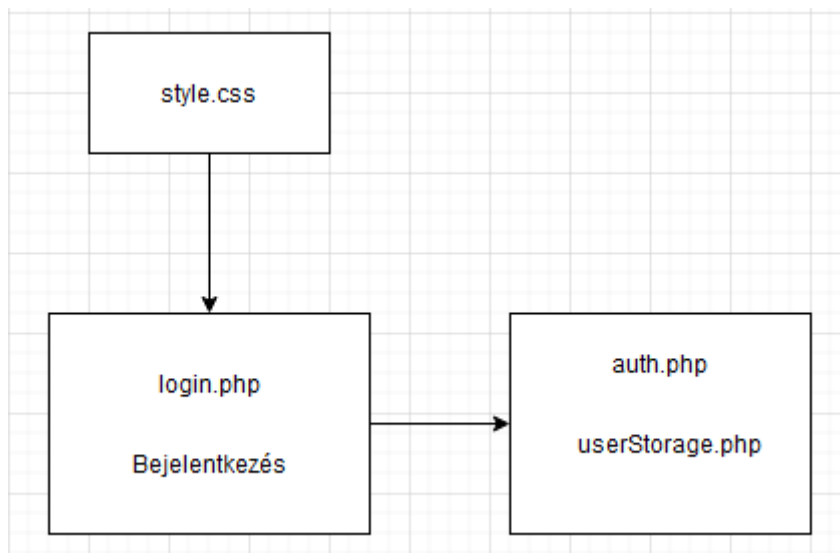
53. ábra Kapcsolat oldal modulkapcsolatok

## Profil



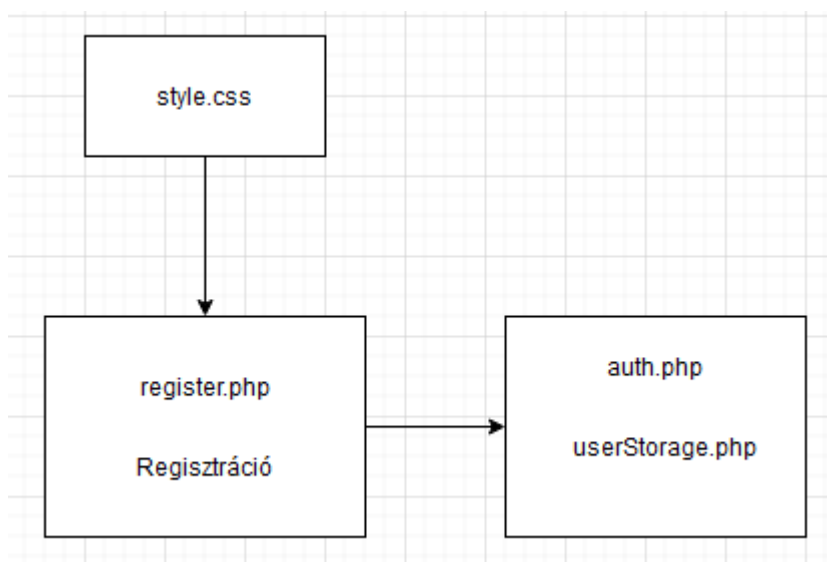
54. ábra Profil oldal modulkapcsolatok

## Bejelentkezés



55. ábra Bejelentkezés oldal modulkapcsolatok

### Regisztráció

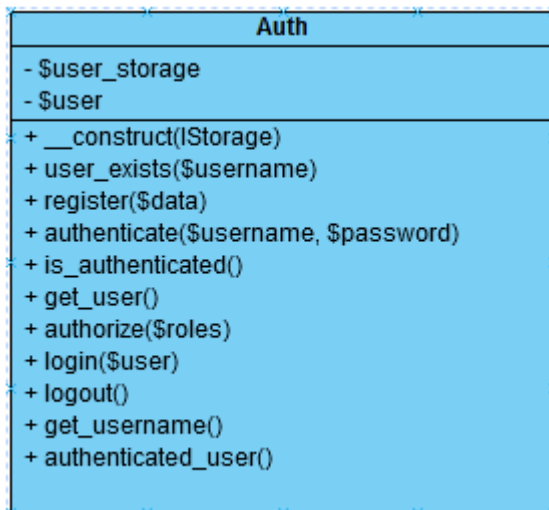


56. ábra Regisztrációs oldal modulkapcsolatok

### 3.3.4 Üzleti logika

#### Adatbázis kezelés a különböző oldalakon

#### Kezdőoldal



57. ábra Auth osztály osztálydiagramja

User	
id	string
username	string
email	string
password	string
roles	string

58. ábra user adattábla

A kezdőoldalon az auth.php file-ban található *is\_authenticated()* függvény segítségével kérdezzük le a *UserStorage*-ból, hogy a felhasználó be van-e jelentkezve. Amikor az oldal betöltődik, akkor létrejön egy Auth osztály, amelynek a konstruktorában a SESSION-ből kiolvassuk, hogy van-e bejelentkezett felhasználó. Az *is\_authenticated()* függvény true-val fog visszatérni, ha volt már bejelentkezett felhasználó, ha nincs bejelentkezett felhasználó, akkor pedig false-szal.

A kezdőoldalon ezt az információt akkor használjuk fel, amikor meghatározzuk, hogy a „Bejelentkezés” gomb szövege Bejelentkezés, vagy Kijelentkezés legyen.

## Játékok oldal

A kezdőoldallal hasonlóan itt is használjuk az *is\_authenticated()* függvényt a „Bejelentkezés” gomb szövegének eldöntésére.



Games	
name	string
type	string
id	string
pic	string
highscores	string
page	string

Az oldalon a játékok felsorolásához le kell kérni a

**GameStorage** osztályból a games.json adatbázis tartalmát, amit a `$gameStorage->getAll()` metódusával tudjuk megtenni. Ez egy az egyben visszaadja a games.json tartalmát. A felsorolásnál lekérdezzük az egyes játékok nevét, típusát, oldalának az url-jét és a játékhoz tartozó képnek az elérési útját.

A felhasználó, ha szűri a játékok felsorolását a megfelelő gomb megnyomásával, akkor az oldal újra fog töltni és a `$_POST` változóból kiolvassa a megfelelő kulcsszót, ami lehet 'akció', 'logikai', 'lövöldözős' vagy 'all', majd meghívja a `chooseCateg()` függvényt, amely felülírja a `$games` változóban található tömböt és csak a megfelelő típusú játékokat hagyja benne. Amennyiben az 'all' kategória lett kiválasztva, akkor az összes játékot be fogja tölteni.

### Kapcsolat oldal

A kezdőoldallal hasonlóan itt is használjuk az `is_authenticated()` függvényt a „Bejelentkezés” gomb szövegének eldöntésére.

A felhasználónak amikor beküldi a form-ot, akkor az oldal újra tölti magát, ekkor az újraindítás után, ha a `$_POST` változónak a hossza nagyobb, mint nulla, akkor az meghívódik a `validate($_POST, $errors)` függvény, ami ellenőrzi, hogy minden mező kitöltésre került-e, ha nem, akkor hozzáadja minden hiányzó adat után a hibaüzenetet az `$errors` tömbhöz. Ha minden rendben volt az adatokkal, akkor elküldi az adatokat a `contactStorage`-nak, ami eltárolja a contacts.json fileban, majd kinullázza a `$_POST` változó elemeit és frissíti az oldalt.

<contacts.json adattábla>

Ha a felhasználó jogosultsága admin szintű, akkor a form alatt kilistázza a contacts.json tartalmát. Innen a nevet, email-t, témát és a szöveget írja ki.

Az admin jogosultságú felhasználóknak van lehetőségük törölni is ezeket a bejegyzéseket. Ehhez meg kell nyomni a törlés gombot. Ekkor a delete.php oldal lesz meghívva. Az URL-ben pedig továbbítva lesz a törlendő adat ID-ja. A delete.php először is az auth.php segítségével eldönti, hogy a felhasználó admin-e az *authorize([„admin”])* metódussal. Majd a **ContactStorage delete(\$contact)** metódusával törli a kért adatot. Ezután visszairányítja a felhasználót a Kapcsolat oldalra.

### Profil oldal

A kezdőoldallal hasonlóan itt is használjuk az *is\_authenticated()* függvényt a „Bejelentkezés” gomb szövegének eldöntésére.

A Játékok oldalhoz hasonlóan itt is le vannak kérve a játékok adatbázis tartalma a \$games változóba. A játékoknál az elérési utat, a nevet és a játékokhoz tartozó kép elérési útját kérjük le. A játékok felsorolásánál még szükség van a felhasználó által elért maximum pontszámra. Ezt úgy tudjuk lekérni, hogy a játékoknál vannak eltárolva az adott játékhoz tartozó összes elmentett pontszám névvel ellátva. A *getGame(\$gameid)* függvény ezt az összes funkciót egybe foglalja. Lekéri a felhasználónevet az **Auth** osztály *get\_username()* metódusával. A \$games változóból végig iterál a highscores tömbön majd ha az egyes elemhez rendelt username megegyezik a most aktív felhasználóval, akkor frissíti a \$maxscore változóban található értéket, amennyiben az magasabb az eddigi legnagyobb pontszámnál. A \$maxscore értéke eredetileg 0, emiatt ha egy adott játékkal még nem játszott a felhasználó, akkor 0-t fog kiírni.

A felhasználó adatait az **Auth** osztály *authenticated\_user()* metódusával töltjük ki, itt a visszatérési értékből kieszedjük a megfelelő helyen a felhasználónevet és az e-mail címet.

### Bejelentkezés oldal

A bejelentkezési oldalon a „Bejelentkezés” gomb lenyomása után az oldal frissítve lesz és a \$\_POST változóba bekerülnek az elküldött adatok. Az oldal frissítés után a *validate(\$P\_POST, &\$data, &\$errors)* függvény ellenőrzi, hogy minden mező kitöltésre

került-e, ha nem, akkor az `$errors` tömböt feltölti a megfelelő hibaüzenetekkel és kiírja a megfelelő mezők mellé, hogy hiányzik az adott adat. Amennyiben a `validate` függvény igazzal tér vissza, akkor az **Auth** osztály `authenticate($data['username'], $data['password'])` metódusával le ellenőrzi, hogy helyesek-e a megadott bejelentkezési adatok. Ha nem helyes adatokkal akar bejelentkezni a felhasználó, akkor az `$errors` tömbhöz hozzáadja a megfelelő hibaüzenetet. Helyes bejelentkezési adatok esetén az **Auth** osztály `login($auth_user)` metódusának a segítségével bejelentkezteti a felhasználót majd visszairányítja a felhasználót a kezdő oldalra.

### Regisztrációs oldal

A regisztrációs oldalon hasonlóan ellenőrzi a megadott adatokat, mint a bejelentkezési oldalon. A `validate($_POST, &$data, &$errors)` ugyanúgy működik, mint a bejelentkezési oldalon. Ha a `validate` függvény igazzal tér vissza, akkor le ellenőrzi, hogy a kívánt felhasználónév vagy e-mail cím már foglalt-e. Amennyiben minden rendben volt az adatokkal, akkor az **Auth** osztály `register($data)` metódusával hozzáadja a felhasználót a **userStorage** `add($user)` metódusával a `users.json` adatbázishoz. Ezek után a felhasználót visszavigálja a bejelentkezési oldalra.

### Játékok adatbázis kezelése

A játékok adatbáziskezelése nagyrészt megegyezik mindegyik játék esetében. A különbség az adatok mennyiségében különbözik, viszont a felhasznált technológia megegyezik.

A kezdőoldallal hasonlóan a játéköldalokon is használjuk az `is_authenticated()` függvényt a „Bejelentkezés” gomb szövegének eldöntésére.

A ranglisták kitöltéséhez be kell töltenünk az adott játék adatait a **GameStorage** osztály segítségével. Ezt hasonlóan tesszük mint a Játékok oldalon is tettük. A betöltött játék esetén lekérjük a `highscores` tömböt amiből a `usort` függvény segítségével sorrendbe tesszük őket a pontjaik alapján csökkenve. Majd az `array_slice` függvénnyel visszaadjuk az 5 legmagasabb elért pontszámot és a hozzátartozó felhasználóneveket.

A játék elmentése funkció az adott játék JavaScript file-jában található. Itt a „Mentés” gomb megnyomásával meghívódik az *onSaveClicked()* függvény amely a játék működéséhez fontos adatokat adja tovább az adott játékhoz tartozó játékmentés oldalnak.

- Köss össze Hármát – a mentés a *save\_connect3.php* oldalon történik ide kerül átadásra a *gamematrix* (tartalmazza az egyes mezők értékeit), *totalSeconds* (a hátralévő idő másodpercben), *points* (a játékos által eddig megszerzett pontok) és *talbesize* (játéktábla mérete)
- Space Shooter – a mentés a *save\_ss.php* oldalon történik, az átadott adatok: *killPoint* (az adott nehézségi szinten mennyit ér egy ellenség megsemmisítése), *enemyProb* (az adott nehézségi szinten milyen valószínűséggel jelennek meg ellenséges űrhajók), *bulletProb* (az adott nehézségi szinten milyen valószínűséggel lőnek az ellenséges űrhajók), *ship* (a felhasználó űrhajójának adatai), *enemies* (az ellenséges űrhajók adatai), *powerups* (a pályán lévő erősítések adatai), *bullets* (a pályán lévő lövedékek adatai), *pressedKey* (milyen irányba mozog éppen az űrhajó), *gameState* (a játék milyen állásban van), *points* (az eddig megszerzett pontok), *gameTime* (az eltelt idő)
- Mini wars – a mentés a *save\_miniwars.php* oldalon történik, az átadott adatok: *points* (az eddig megszerzett pontok), *gameTime* (az eddig eltelt idő), *gamebases* (a pályán lévő bázisokhoz tartozó adatok), *gamepath* (a bázisokat összekötő utak adatai)
- Kártyás memória játék – a mentés a *save\_memorygame.php* oldalon történik, az átadott adatok: *gamematrix* (a pálya mezőinek az adatai), *totalSeconds* (az

eddig eltelt idő), points (az eddig megszerzett pontok), tabelsize (a pálya mérete), revealedarr (az eddig megtalált párok sorszáma)

A mentés oldalon a **GameSavesStorage** osztály *add(\$savedgame)* metódusának a segítségével hozzá lesz adva a gamesaves.json adatbázishoz a mentett játék.

<gamesaves.json adattábla>

A játék betöltése az adott játék php oldalán kezdődik, úgy, hogy először a **GameSavesStorage** osztály *getAll()* metódusával lekérjük az összes elmentett játékot és ezt bemásoljuk egy „div” -be aminek az értékét a JavaScript oldalon beolvassuk a *loadJSON()* függvény segítségével. Ott kiválogatja a megfelelő játékot és azon belül is a megfelelő felhasználóhoz tartozó mentéseket, amik alapján megkonstruálja a betöltési oldalt, amihez a „Betöltés” gomb lenyomásával tud hozzáférni a felhasználó. A megfelelő felhasználót úgy tudja kiválasztani a kód, hogy az adott játék php oldalán egy div-be bemásolja az aktív felhasználóhoz tartozó felhasználói ID-t, amit a JavaScript kód kiolvas ebből a div-ből.

Bejelentkezett felhasználó esetén amikor a játéknak vége, akkor meghívódik a *postScore()* függvény, ami meghívja az adott játékhoz tartozó eredményelmentő php oldalt. Az egyes játékokhoz tartozó oldalak nevei:

- Köss Össze Hármat – *postscore\_connect3.php*
- Space Shooter – *postscore\_spaceshooter.php*
- Mini Wars – *postscore\_miniwars.php*
- Kártyás memória játék – *postscore\_memorygame.php*

Ezek az oldalakon lekérjük az aktív felhasználó felhasználónevét az **Auth** osztály *get\_username()* metódusával, majd a **GameStorage** osztály *update()* metódusával kiegészítjük az adott játékhoz tartozó highscores tömbjét.

## Játékok üzleti logikája

### Space shooter

A Space Shooter játék betöltésekor az oldalon található mentés és szünet gombot deaktiváljuk. Amennyiben a felhasználó nincs bejelentkezve, akkor a betöltés gombot szintén deaktiváljuk.

Ha a felhasználó elindítja a játékot a „Start” gomb megnyomásával, akkor az *onStartClicked()* függvény kerül meghívásra és a játék canvas felfedésre kerül. Ez a függvény állítja be a kiválasztott játékszint alapján a háttérben a játékos által irányított űrhajó adatait, az ellenség megjelenésének valószínűségét, azt, hogy mekkora az esélye, hogy az ellenséges űrhajó milyen eséllyel fog lövedéket kilőni és, hogy mennyi pontot kap a játékos minden elpusztított ellenség után. Majd elindítja a *timePass()* és a *gameLoop()* függvényt.

A *timePass()* függvény növeli az időt minden másodpercben.

A *gameLoop()* függvény számon tartja az eltelt időt, és ez alapján meghívja az *update()* és *draw()* függvényt.

Az *update()* függvény frissíti az eltelt idő alapján a játékos által irányított űrhajó, a pályán lévő ellenséges űrhajók és a pályán lévő erősítések pozícióját a játéktérben, valamint ellenséget is létrehozhat, a *Math.random()* függvény alapján. Ezekon kívül ellenőrzi, hogy volt-e ütközés a játékos űrhajója és egy ellenséges űrhajó vagy egy erősítés vagy egy lövedék között, ha volt ütközés ellenség és a játékos űrhajója között, akkor kinullázza a játékos életerejét, ha erősítéssel volt ütközés, akkor azt aktiválja, ha egy ellenség által kilőtt lövedékkel ütközik, akkor levon egy életet. Továbbá azt is megvizsgálja, hogy volt-e ütközés ellenség és a játékos által kilőtt lövedék között, ha igen, akkor az ellenség megsemmisül. Végül pedig ellenőrzi, hogy vége van-e a játéknak, az alapján, hogy van-e még élete a játékosnak.

A *draw()* függvény felelős a canvason látható grafikák frissítésére.

Ha a játékos lenyomja a balra nyilat vagy a jobbra nyilat a billentyűzetén, akkor ezt érzékeli az *onKeyDown(event)* függvény és beállítja az úrhajó menetirányát a megfelelő irányba. Továbbá az is érzékeli, hogy ha a játékos lenyomja az 'X' gombot, ekkor lövedéket indít a játékos úrhajójának pillanatnyi pozíciójából.

Ha a játékos felengedi az eddig lenyomva tartott balra vagy jobbra nyilat a billentyűzetén, akkor az *onKeyUp(event)* függvény megállítja az úrhajó eddigi mozgását.

Az *onPauseClicked()* függvény akkor hívódik meg, amikor a felhasználó megnyomja a „Szünet” gombot. Ekkor ha eddig a játék folyamatban volt akkor szünetelteti a játékot. Ha eddig szünetelve volt a játék akkor pedig elindítja a szünet gomb megnyomása előtti állapotból.

Az *onSaveClicked()* függvény leírását a **Játékok adatbázis kezelése** pontban már részleteztem.

Ha a felhasználó megnyomja a „Betöltés” gombot, akkor a játékfelület eltűnik és megjelenik a betöltési oldal, itt a **Játékok adatbázis kezelése** pontban már részleteztem, hogy hogyan kapja meg a mentett játékokat a JavaScript kód, ezért ezt már itt nem fogom részletezni. A mentett játékokat a *loadJSON()* függvény fogja kirajzolni a betöltési képernyőre. Ez a függvény létrehoz annyi canvas-t amennyi mentés volt a felhasználó nevével ellátva, ezeknek a kirajzolása a *drawMiniCanvas(minicanvas, data)* függvényben történik.

A *drawMiniCanvas* függvény hasonlóan működik mint a *draw()* függvény, annyi különbséggel, hogy itt paraméterként kapja meg, hogy melyik canvasra (minicanvas) milyen adatokkal ellátott játékot (data) rajzoljon ki. A *loadcanvasloop()* minden másodpercben meghívódik, hogy a kirajzolt canvasok képe megmaradjon mindaddig amíg a felhasználó nem választ ki egy betölteni kívánt játékot.

A felhasználó úgy tud kiválasztani egy betölteni kívánt mentést, hogy rákattint egy betöltési oldalon lévő canvasra. Ezt a kattintást az *onPageClicked(event)* függvény figyeli. A kattintás során ez a függvény kéri le, hogy milyen ID-vel rendelkező canvast

választott ki a felhasználó, majd ez alapján az ID alapján meghívja a *loadNewGame(id)* függvényt.

A *loadNewGame(id)* függvény frissíti a játékhoz szükséges adatokat, majd elindítja az új játékot.

Ha a felhasználó megnyomja a „Játékleírás” gombot, akkor az *onDescriptionClicked()* függvény láthatóvá teszi a Játékleírást. Ismételt lenyomás után eltűnteti a leírást.

### **Köss össze hármat**

A Köss össze hármat játék betöltésekor az oldalon található mentés és szünet gombot deaktiváljuk. Amennyiben a felhasználó nincs bejelentkezve, akkor a betöltés gombot szintén deaktiváljuk.

Ha a felhasználó elindítja a játékot a „Start” gomb megnyomásával, akkor az *onStartClicked()* függvény kerül meghívásra és a játékfelület láthatóvá válik. Ebben a függvényben inicializáljuk a játék elemeit.

A *generateRandomMatrix()* tölti fel random elemekkel a játéktáblát. Itt a létrejött random táblára meghívja a *checkMatrix(gamematrix)* függvényt, ami megvizsgálja a játéktáblát, hogy van-e 3, 4 vagy 5 azonos értékű mező egymás mellett függőlegesen és vízszintesen. Ha talál ilyet, akkor a *refill(gamematrix)* függvény kicseréli random értékekkel ezeket mezőket mindaddig, amíg szerepel ez a kombináció valahol a játéktáblában.

A *generateTable(tablesize)* függvény létrehozza a látható táblázatot és a gamematrix alapján megjeleníti benne az elemeket.

Az *addEventListener()* függvény hozzárendeli minden mezőhöz a *getClickedCell()* event listenert, ami segít beazonosítani a kiválasztott mezőt.

A *setTime()* függvény felelős az idő múlására és idő alapján a játék végének meghatározására is.

Ha a játékos kattintással kiválaszt egy mezőt, akkor a *getClickedCell(event)* függvény meghatározza melyik mezőt nyomta meg a felhasználó, majd ha még nem választott ki



ezelőtt elemet, akkor eltárolja a választást egy *firstChosen* nevű változóba és a kiválasztott mező képét lecseréli egy olyanra, aminek szürke a háttere ezzel is indikálva a kiválasztott mezőt. Ha másodszor is kiválaszt egy mezőt a játékos, akkor a második választás el lesz tárolva a *secondChosen* változóba és a *moved(firstChosen, secondChosen)* függvény kerül meghívásra.

A *moved* függvény felelős azért, hogy megvizsgálja, hogy egy érvényes lépés, volt-e amit a játékos próbált tenni, ha nem, akkor változatlanul hagyja a pályát, ha igen, akkor felcseréli a két elemet és sárgává változtatja az így kapott egymás mellett egy sorban vagy oszlopban lévő objektumok hátterét, majd egy másodperc múlva kiveszi őket a játéktáblából és feltölti a helyeiket a fölöttük lévő mezők értékeivel, végül a tábla tetejére új elemeket generál és frissíti a látható játékkeret.

Ha a játékos megnyomja az „Újratöltés” gombot, akkor az egész játéktábla le lesz cserélve a *generateRandomMatrix* függvényhívással.

Az *onSaveClicked*, *onPauseClicked*, *onLoadClicked*, *onDescriptionClicked*, *onPageClicked* és *loadNewGame* függvények hasonlóan működnek, mint a Space Shooter játéknál, ezért ezt itt most nem fejtem ki külön.

### **Mini wars**

A Mini wars játék betöltésekor az oldalon található újrakezdés, mentés és szünet gombot deaktiváljuk. Amennyiben a felhasználó nincs bejelentkezve, akkor a betöltés gombot szintén deaktiváljuk.

Az oldal betöltésekor a pályaválasztás oldal jelenik meg a felhasználó előtt, ezt a *drawChooseGame()* függvény hozza létre és meghívja a *drawStartupCanvas()* függvényt.

A *drawStartupCanvas()* kirajzolja az összes lehetséges pályát amik az allgames tömbben vannak elhelyezve.

Az *onPageClicked(event)* függvény érzékeli ha az oldalon kattintás történt, ha a pályaválasztás oldal az aktív, akkor kiválasztott pálya esetén meghívja a

*loadNewStartupGame()* függvényt, ha a betöltés oldal aktív, akkor pedig a *loadNewGame()* függvény kerül meghívásra.

A *loadNewStartupGame* függvény inicializálja a játékot a kezdőértékekkel és bemásolja a gamebases és gamepath tömbökbe a kiválasztott pálya adatait, majd meghívja a *draw()* és *timePass()* függvényeket.

A *draw()* függvény kirajzolja a játékfelületre a megfelelő pályát.

A *timePass()* függvény felelős az idő múlásának számontartására, a *checkEnd()* és *enemyAttack()* függvények meghívására.

A *checkEnd()* függvény ellenőrzi, hogy a játékos vagy az ellenség elfoglalta-e az összes bázist, ami a pályán található, ha igen, akkor vége a játéknak.

Az *enemyAttack()* végig iterál a bázisokon és ha talál egyet, amit az ellenség birtokol, akkor van esélye annak, hogy az a bázis támadást indítson egy másik olyan bázis irányába, ahova az adott ellenséges bázisból vezet út, az *attack()* függvény meghívásával.

Az *attack()* függvény paraméterül kap két bázist. Az első a támadó a második a védekező.

Ha a játékos rákattint a canvasra, azt a canvashoz kötött event listener fogja érzékelni, ami meghívja a *getCursorPosition(canvas, e)* függvényt.

A *getCursorPosition* függvény a *baseClicked(x,y)* függvény segítségével meghatározza, hogy melyik bázist választotta ki a játékos. majd ha saját bázisára kattintott, akkor a bázis ID-ját értékül adja a firstchosen változónak. Másodszori bázisválasztás után a függvény meghívja az *attack* függvényt, majd a *draw()* függvényt.

Ennél a játéknál az *onStartClicked()*, *onPauseClicked()*, *onSaveClicked()*, *loadcanvasloop()*, *onLoadClicked()*, *drawMiniCanvas()*, *loadNewGame()*, *loadJson()* és *onDescriptionClicked()* ugyanazt csinálja mint a Space Shooter játéknál, emiatt ezt nem fogom még egyszer leírni.

## Kártyás memóriajáték

A Mini wars játék betöltésekor az oldalon található mentés és szünet gombot deaktiváljuk. Amennyiben a felhasználó nincs bejelentkezve, akkor a betöltés gombot szintén deaktiváljuk.

Ennél a játéknál az *onStartClicked()*, *generateRandomMatrix()*, *generateTable()*, *setTime()* és *addEventListener()* függvények ugyanazt csinálják, mint a Köss Össze Hármát játéknál, emiatt azt nem fogom még egyszer leírni.

A *getClickedCell(event)* függvény abban különbözik a Köss Össze Hármát játéktól, hogy itt mindkét kiválasztás esetén meghívódik a *revealCard(row, col)* függvény, majd a második kiválasztás után a *checkPair()* függvény is.

A *revealCard* függvény felfedi a kiválasztott mező értékét.

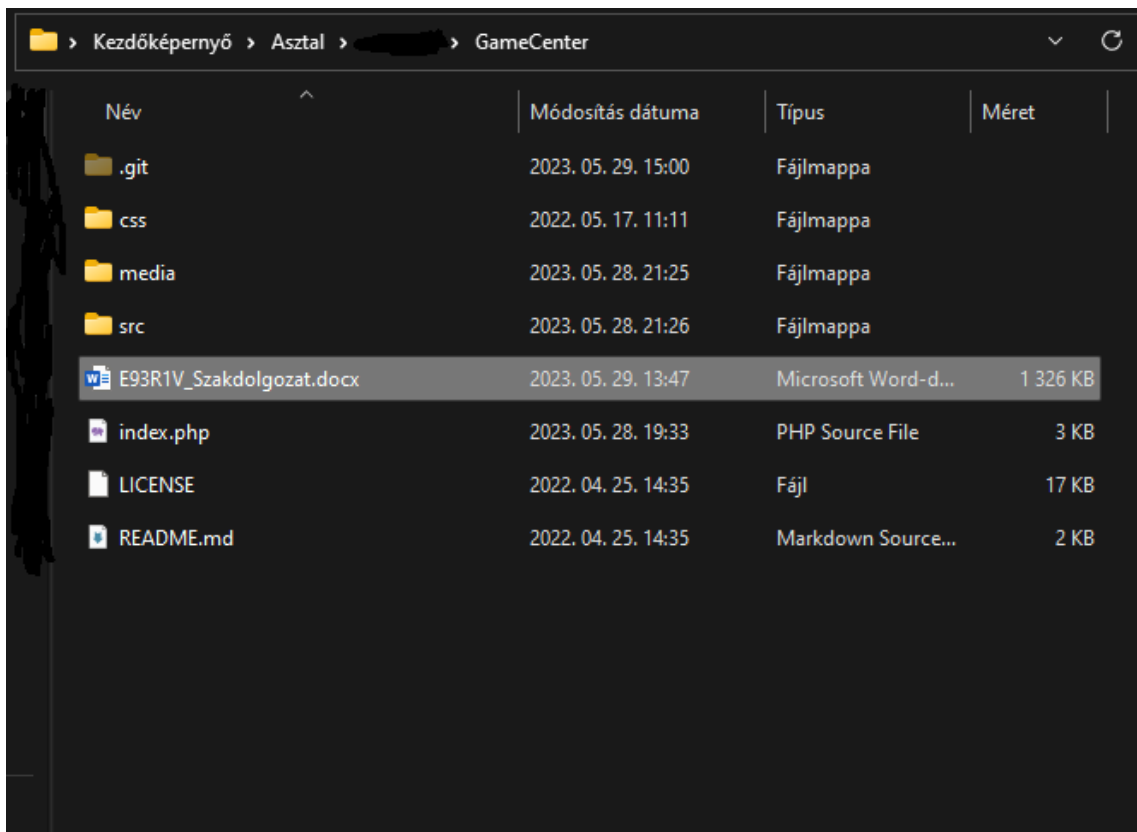
A *checkPair* függvény le ellenőrzi, hogy a két kiválasztott mező értéke ugyanaz-e. Ha igen, akkor a kártyák felfordítva maradnak, ha nem, akkor visszafordul mindkét kártya a *closeChosen()* függvényhívás hatására. Továbbá ez a függvény ellenőrzi, hogy vége van-e a játéknak.

Az *onPauseClicked()*, *onSaveClicked()*, *onLoadClicked()*, *loadJson()*, *onPageClicked()*, *loadNewGame()* és *onDescriptionClicked()* függvények megegyeznek a Köss Össze Hármát játékkal.

## 3.4 Futtatás

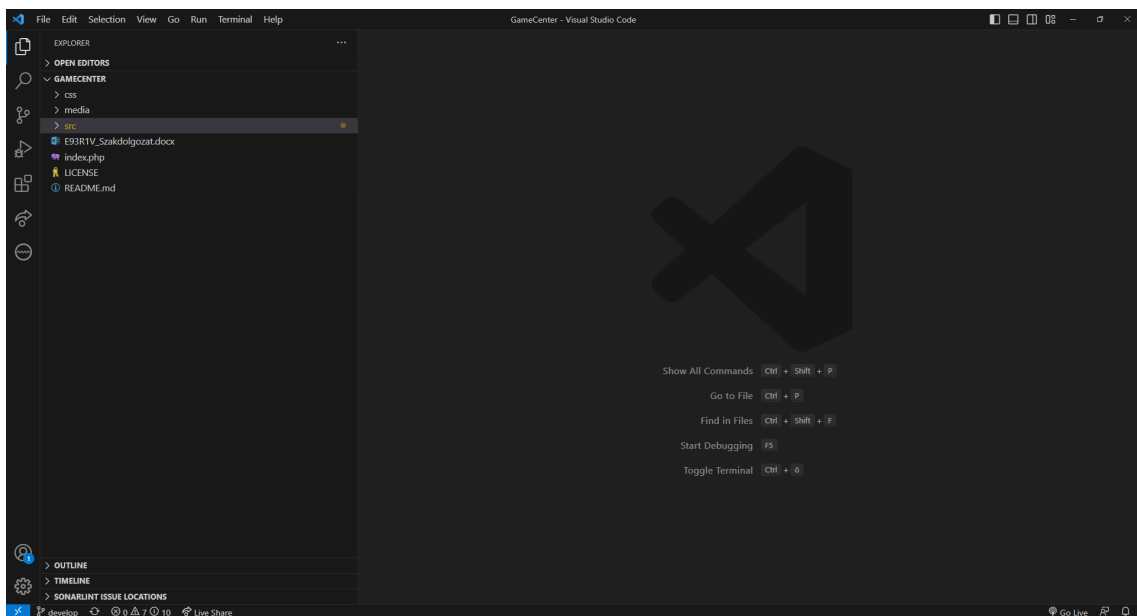
Itt fogom részletezni, hogy hogyan lehet elindítani a kitömörített fájlból a weboldalt.

Kicsomagolás után a következő mappa fog megjelenni:



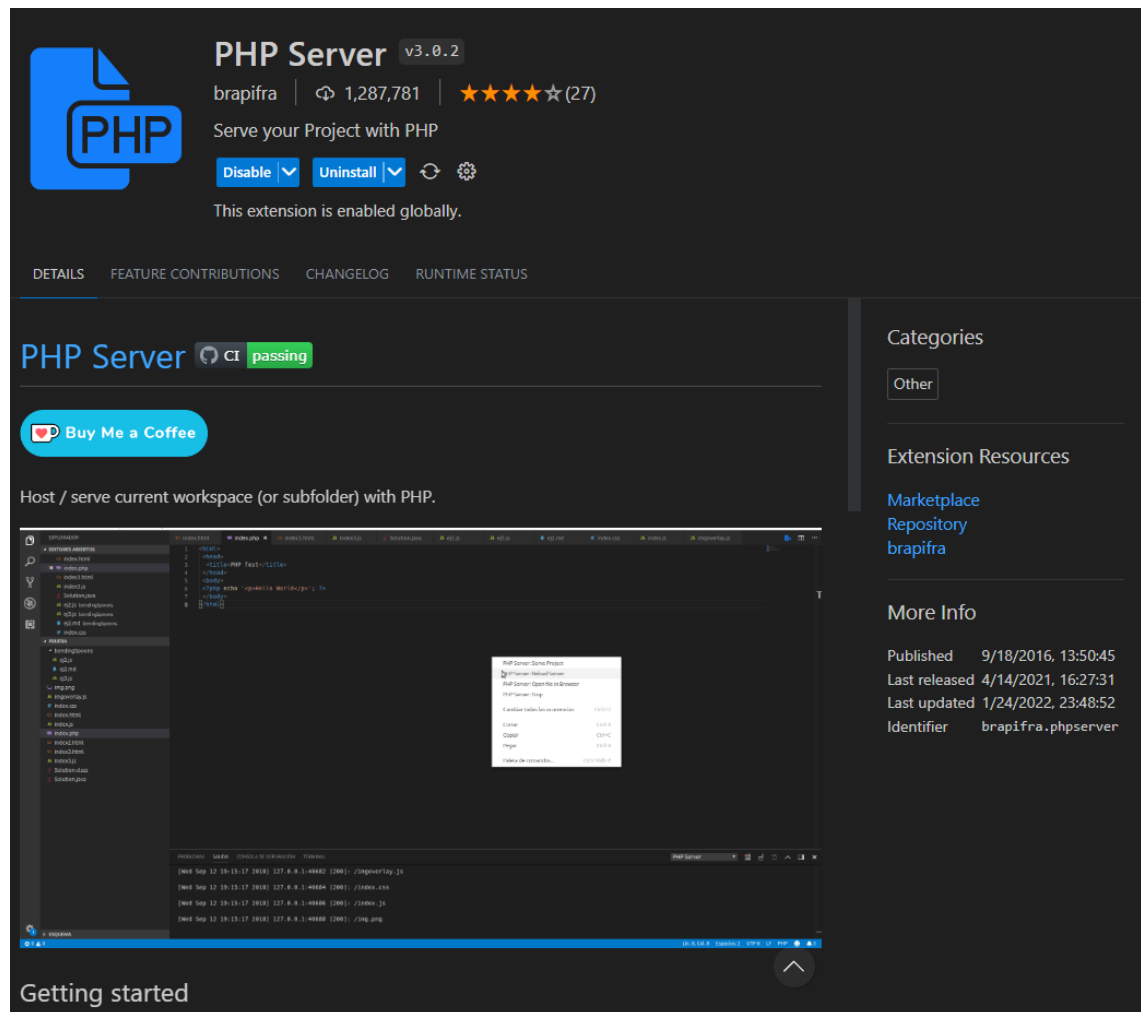
59. ábra Kicsomagolt mappa

Ezt a mappát kell megnyitni Visual Studio Code segítségével.



60. ábra Mappa megnyitva Visual Studio Code-ban

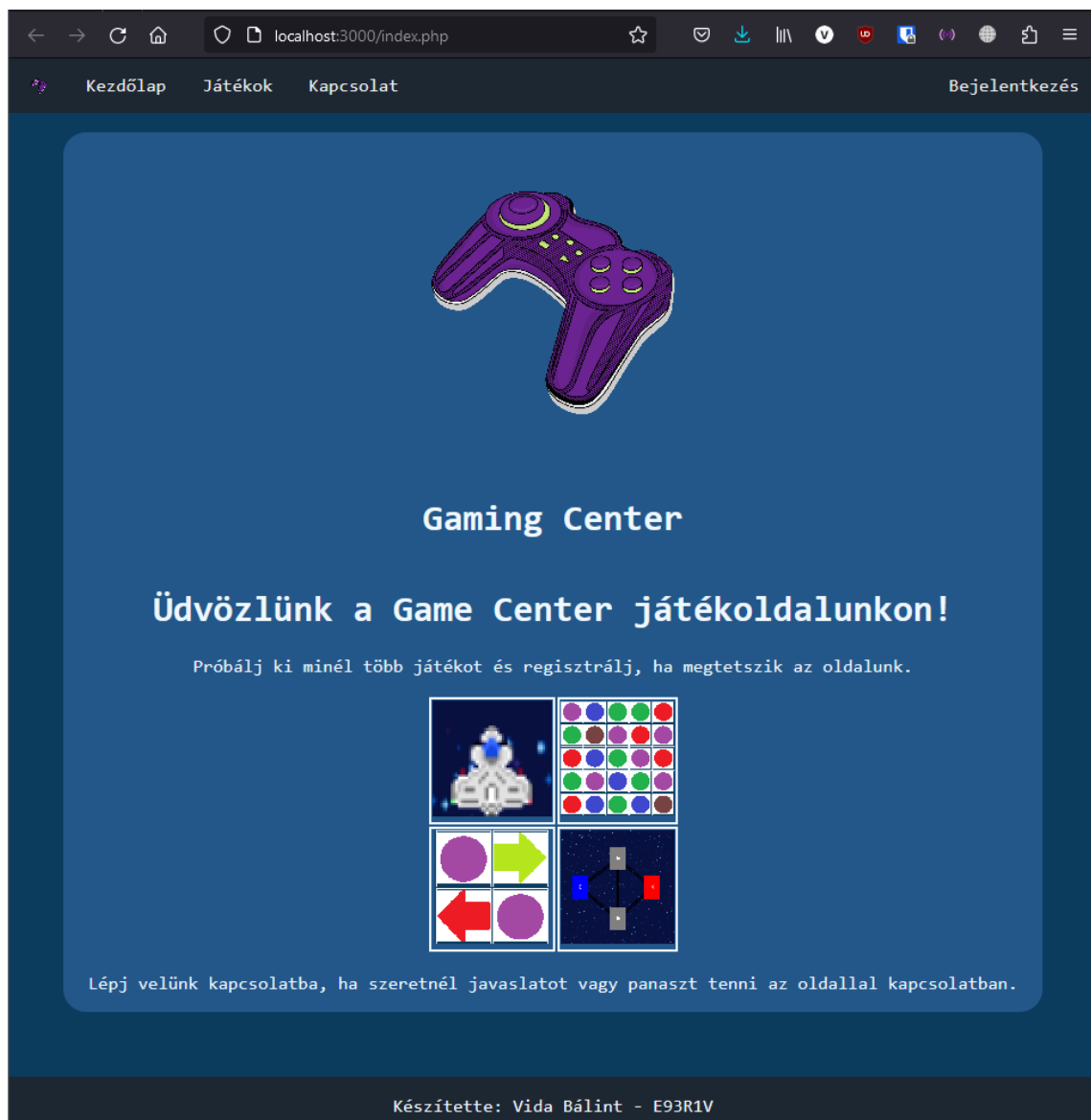
A Visual Studio Code-ban szükség lesz egy extension-re, amely lehetővé teszi lokális környezetben egy PHP szerver indítását. Erre én a PHP Server extension-t használtam:



61. ábra PHP szerver

Miután ez installálva lett meg kell nyitni az index.php file-t a mappán belül.





64. ábra megjelenik a weboldal

### 3.5 Tesztelés

A teszteket a 3.2.3-as User-story diagram alapján végeztem el.

Mint	Adott	Amikor	Akkor	Teszt
USER	Weboldal be van töltve	Navigációs fül egyik gombjára kattint	Betölt a megfelelő oldal	OK
USER	Weboldal be van töltve	Bejelentkezés/Regisztráció gombra kattint	Betölt a bejelentkezés/regisztrációs oldal	OK
USER	Bejelentkezés/regisztrációs oldal be van töltve	Helytelen adatokkal próbál belépni	Hibaüzenet megjelenik a kitöltendő szövegmezők alatt és	OK

Mint	Adott	Amikor	Akkor	Teszt
			nem történik bejelentkezés	
USER	Bejelentkezés/ regisztrációs oldal be van töltve	Helyes adatokkal próbál belépni	Bejelentkezés megtörténik, megjelenik a Profil fül a navigációs sávban	OK
USER	Bejelentkezés/ regisztrációs oldal be van töltve	Regisztrálni próbál helytelen adatokkal	Hibaüzenet megjelenik a kitöltendő szövegmezők alatt és nem történik regisztráció	OK
USER	Bejelentkezés/ regisztrációs oldal be van töltve	Regisztrálni próbál helyes adatokkal	Regisztráció megtörténik, bejelentkezés megtörténik, megjelenik a Profil fül a navigációs sávban	OK
USER	Weboldal be van töltve	Megnyomja a kijelentkezés gombot	Kijelentkezés megtörténik, eltűnik a Profil fül a navigációs sávban	OK
USER	Profil oldal be van töltve		Megjelenik egy táblázatban az összes játékhoz tartozó egyéni legjobb eredménye	OK
USER	Kapcsolat oldal be van töltve	Üresen hagyja a kitöltendő mezőket	Hibaüzenet megjelenik a kitöltendő szövegmezők alatt és nem történik elküldés	OK
USER	Kapcsolat oldal be van töltve	Kitölti a megfelelő mezőket helyes adatokkal majd próbálja elküldeni	Megtörténik az elküldés	OK
USER	Játék kiválasztás oldal be van töltve	Kiválasztja a szűrési tulajdonságokat majd rányom a Szűrés gombra	A szűrésnek megfelelő játékok megjelennek	OK
USER	Játék kiválasztás oldal be van töltve	Kiválaszt egy játékot kattintással	Betölt az adott játék a neki megfelelő oldalon	OK
USER	Játékoldal be van töltve a megfelelő játékkal	Megnyomja a Játékleírás gombot	Megjelenik az adott játék leírása és a kezelési leírása	OK



Mint	Adott	Amikor	Akkor	Teszt
USER	Játékoldal be van töltve a megfelelő játékkal	Elkezd játszani	A játék elindul és megfelelően működik	OK
USER	Játék oldal be van töltve a megfelelő játékkal, be van jelentkezve a felhasználó	Mentés gombra kattint	A játék elmentésre kerül a felhasználó neve alatt	OK
USER	Játékoldal be van töltve a megfelelő játékkal, nincs bejelentkezve a felhasználó	Mentés gombra kattint	Nem történik semmi	OK
USER	Játék oldal be van töltve a megfelelő játékkal, be van jelentkezve a felhasználó	Betöltés gombra kattint	Megjelennek a betölthető játékok, ha vannak ilyenek	OK
USER	Játék oldal be van töltve a megfelelő játékkal, be van jelentkezve a felhasználó, megjelennek a betölthető játékok	Rákattint az egyik betölthető játékra	Betöltődik a játékállás és onnan folytathatja a játékot	OK
USER	Játék oldal be van töltve a megfelelő játékkal, a játék folyamatban van	Rányom a játék szüneteltetése gombra	A játék állása szünetel, az idő mérése leáll	OK
USER	Játék szünetel	Próbál tovább játszani	Nem történik semmi	OK
USER	Játék szünetel	Megnyomja a játék folytatása gombot	Folytatódik a játék, az idő mérése folytatódik	OK

Mint	Adott	Amikor	Akkor	Teszt
USER	Játszik a felhasználó	Megnyomja az új játék gombot	Megkérdezi az oldal, hogy szeretné-e elmenteni az aktuális játékot	OK

#### Space Shooter:

Adott	Amikor	Akkor	Teszt
A játék indul.	A játékos megnyomja a "start" gombot.	A játék elkezdődik, és a játékos űrhajója megjelenik a képernyőn.	OK
A játék folyamatban van	A játékos megnyomja az "lövés" gombot.	Egy lézer sugár indul az űrhajóból az ellenséges űrhajók felé.	OK
A játék folyamatban van	A játékos a nyíl billentyűvel mozgatja az űrhajót.	A játékos űrhajója a képernyőn kívánt irányba mozog.	OK
A játék folyamatban van	A játékos felvesz egy erősítést.	Az űrhajó új képességet vagy erőt kap, például elpusztíthatatlan lesz.	OK
A játék folyamatban van	Az űrhajó összeütközik egy ellenséges űrhajóval vagy akadállyal.	Az űrhajó életet, az ellenséges űrhajó vagy aszteroida pedig elpusztul.	OK

Adott	Amikor	Akkor	Teszt
Egy ellenséges űrhajó lő egyet		Egy lézer sugár indul az ellenséges űrhajóból a játékos űrhajója felé.	OK

#### Mini Wars:

Adott	Amikor	Akkor	Teszt
A játék indul.		Az egységtermelés elindul minden játékos vagy ellenség által birtokolt mezőn.	OK
A játék folyamatban van	A játékos kiválaszt először egy általa birtokolt mezőt, majd egy másik általa birtokolt mezőt	Az először kiválasztott mezőből az összes egység átkerül a másodjára kiválasztott mezőre.	OK
A játék folyamatban van.	A játékos kiválaszt először egy általa kiválasztott mezőt, majd egy ellenséges vagy senki által nem birtokolt mezőt.	A játékos által birtokolt mezőről az összes egység támadni kezdi a másodjára kiválasztott mezőt.	OK
A játékos megtámadott egy ellenséges vagy senki által nem birtokolt mezőt, a küldött csapatok száma kisebb, mint amennyien védik a mezőt.		A védekező mező nem lesz elfoglalva, annyival kevesebb védő marad, mint ahány egységgel a játékos támadott.	OK
A játékos megtámadott egy ellenséges vagy senki által nem birtokolt mezőt, a		A megtámadott mezőt már a játékos birtokolja.	OK

Adott	Amikor	Akkor	Teszt
küldött csapatok száma nagyobb, mint amennyien védik a mezőt.			
Egy ellenséges mezőről támadás indul egy senki által nem birtokolt mezőre vagy a játékos által birtokolt mezőre, a küldött csapatok száma kisebb, mint amennyien védik a mezőt.		A védekező mező nem lesz elfoglalva, annyival kevesebb védő marad, mint ahány egységgel az ellenfél támadott.	OK
Egy ellenséges mezőről támadás indul egy senki által nem birtokolt mezőre vagy a játékos által birtokolt mezőre, a küldött csapatok száma nagyobb, mint amennyien védik a mezőt.		A megtámadott mezőt már az ellenfél birtokolja.	OK
A játékos vagy az ellenfél elfoglalta az összes mezőt.		A játéknak vége, a játékos megkapja a végső pontszámát.	OK

Köss Össze Hármat:

Adott	Amikor	Akkor	Teszt
A játék folyamatban van	A felhasználó két szomszédos kört kiválaszt, hogy felcserélje őket	Ha az áthelyezés egy három vagy sor, vagy oszlop kialakulásához vezet, akkor ezek a tárgyak eltűnnek, és a velük fentről követő körök leesnek. Új körök töltik fel a rács tetején lévő hézagokat.	OK
A játék folyamatban van	A felhasználó kiválaszt egy kört	Annak a körnek a háttere szürke lesz, jelezve, hogy az a felhasználó kiválasztott köre.	OK

Adott	Amikor	Akkor	Teszt
A játék folyamatban van	Lejár az idő	A játék véget ér, és a végső pontszámot megjelenítik.	OK
A játék folyamatban van	A felhasználó kiválaszt egy kört, majd egy másik kört, amelyet nem lehet az elsővel felcserélni	Semmi sem történik.	OK
A játék folyamatban van	A felhasználó kicserél két kört, melynek következtében 3, 4 vagy 5 kör lesz egymás mellett egy sorban vagy oszlopban.	Eltűnik mindegyik kör, majd fel lesz töltve újra	OK

Kártyás memória játék:

Adott	Amikor	Akkor	Teszt
A játék indul.	A játékos megnyomja a "start" gombot.	A játék elkezdődik, és a kártyák le vannak fordítva.	OK
A kártyák le vannak fordítva.	A játékos kattint egy kártyára.	A kártya megfordul, és az ikonja látható lesz.	OK
Egy kártya látható az ikonjával.	A játékos kattint egy másik kártyára.	Ha az ikonok megegyeznek, a két kártya eltűnik, és a játékos pontot kap. Ha nem, akkor mindkét kártya visszafordul.	OK
A játék folyamatban van	A játékos megnyomja a "új játék" gombot.	Az összes kártya visszafordul, és a játék újra kezdődik.	OK
A játék folyamatban van	A játékos megnyomja a	A játék szünetel, visszatérhet a játékhoz.	OK

Adott	Amikor	Akkor	Teszt
	"szünet" gombot.		
A játék idő lejár	A játékosnak nem sikerült megtalálnia az összes párt	A játéknak vége.	OK
A játékost megtalálta az összes párt		Megáll az időzítő, a játékos a maradék idő alapján kap extra pontokat.	OK

## 4 Összefoglalás

A honlap megvalósítása teljesíti az elvárt követelményeket. Úgy érzem, hogy sikeresen véghez tudtam vinni a gyerekkori álmomat, és nekem is sikerült egy weboldalt csinálnom, ahol játékokkal lehet játszani.

Bár voltak a megvalósításban némi akadályok, amiket le kellett küzdenem, például egy ilyen komplex kész projektet még soha nem kellett csinálnom, emiatt az is kihívás volt, hogy hogyan tervezzem meg, és a részletekben sem szabad nagyon elveszni tervezés során, mert amikor a tényleges megvalósításra kerül a sor, akkor úgy is változni fog minden apróbb részlet a tervezettekhez képest.

Sikerült immár sok tapasztalatot gyűjtenem egy ilyen komplexebb feladat megoldásában, és a webfejlesztési készségeimet is sikerült egy magasabb szintre emelnem.

### 4.1 Továbbfejlesztési lehetőségek

A továbbfejlesztés két főbb irányban tudom elképzelni. Egyrészt sql alapú adatbázist be lehetne vezetni az adatok tartós tárolására a json alapú adatbázis helyett. Másrészt a játékokkal kapcsolatban lehetne még több és több játékot hozzáadni. A játékok értékelése is egy hasznos funkció lehetne.

## 5 Források

### 5.1 Felhasznált szerkesztőprogramok

[1] Draw IO online diagramkészítő oldal: <https://app.diagrams.net/>

[2] Visual Paradigm online diagramkészítő: <https://online.visual-paradigm.com/>

[3] MockFlow online drótvázterv készítő: <https://mockflow.com/>

[4] DBDiagram online adatbázis tervező: <https://dbdiagram.io/home>

[5] Visual Studio Code szoftver: <https://code.visualstudio.com/>

## 5.2 Felhasznált szakirodalom

[6] W3Schools tutorial: <https://www.w3schools.com/>

[7] Stack Overflow: <https://stackoverflow.com/>

[8] Horváth Győző, Visnovitz Márton - Dinamikus weboldalak előállítása szerveroldali technológiákkal: <http://webprogramozas.inf.elte.hu/tananyag/szerver/>

## 5.3 Felhasznált dokumentáció

[9] Wikipedia online enciklopédia, JavaScript dokumentáció:  
<https://hu.wikipedia.org/wiki/JavaScript>

[10] Wikipedia online enciklopédia, PHP dokumentáció:  
<https://hu.wikipedia.org/wiki/PHP>

[11] Wikipedia online enciklopédia, CSS dokumentáció:  
<https://hu.wikipedia.org/wiki/CSS>

[12] Wikipedia online enciklopédia, HTML5 dokumentáció:  
<https://hu.wikipedia.org/wiki/HTML5>

## 5.4 Egyéb

[13] Vecteezy stock image hub: <https://www.vecteezy.com/png/15370017-gaming-console-illustration-isolated-on-png-transparent-background>