Szakképesítés megnevezése: **Szoftverfejlesztő**

OKJ száma: **54 213 05**

ZÁRÓDOLGOZAT

Kosárlabda kártyajáték

|  |  |
| --- | --- |
| Györgyi Tamás | Vorák Gergő |
| témavezető | 14.S |

Budapest, 2020

# Nyilatkozat

Alulírott, Vorák Gergő kijelentem, hogy ez a záródolgozat saját tudásom, önálló munkám terméke.

Kijelentem, hogy a záródolgozat beköttetett és elektronikus formában leadott példányai mind formátumban, mind tartalomban egyezőek, eltérést nem tartalmaznak.

Dátum:

hallgató aláírása

# Tartalomjegyzék

[Nyilatkozat 2](#_Toc35421984)

[Tartalomjegyzék 3](#_Toc35421985)

[1. Bevezetés 4](#_Toc35421986)

[2. A témaválasztás indoklása 6](#_Toc35421987)

[3. Fejlesztői dokumentáció 8](#_Toc35421988)

[3.1. Az alkalmazott fejlesztői eszközök 8](#_Toc35421989)

[3.2. Felületterv 9](#_Toc35421990)

[3.3. Adatmodell leírása 14](#_Toc35421991)

[3.4. Részletes feladatspecifikáció, algoritmusok 16](#_Toc35421992)

[3.5. Tesztelési dokumentáció 21](#_Toc35421993)

[3.6. Továbbfejlesztési lehetőségek 22](#_Toc35421994)

[4. Felhasználói dokumentáció 23](#_Toc35421995)

[4.1. A program általános specifikációja 23](#_Toc35421996)

[4.2. Rendszerkövetelmények 25](#_Toc35421997)

[4.3. A program telepítésének és konfigurálásának a leírása 26](#_Toc35421998)

[4.4. A program használatának a részletes leírása 27](#_Toc35421999)

[5. Összegzés 28](#_Toc35422000)

# 1. Bevezetés

A szoftverfejlesztő OKJ-s képzés elvégzéséhez szükséges elkészítendő szakdolgozatom tervét szeretném ismertetni. A szakdolgozatomat webes formában szeretném megvalósítani, témám az amerikai kosárlabda ligához (NBA) kapcsolódik. Szeretnék megvalósítani egy statisztikai oldalt, mivel jól elkészített magyar nyelvű oldal nem található a témában. Ehhez szeretnék hozzákapcsolni egy kártyás játékot is. Ezáltal a rajongók egy helyen meg tudják nézni a játékosok teljesítményét, majd játszhatnak.

A kinézetre nagy hangsúlyt szeretnék fektetni az projektem megvalósításánál, mivel a frontend kiemelten érdekel. Emellett az adatbázis megtervezése és megvalósítása is fontos része a projektemnek, mivel a csapatok és játékosok adatai ezáltal válhatnak hozzáférhetővé.

Mivel IT szakember szeretnék lenni, így nagyon fontos számomra a megfelelő jogosultságkezelés. 3 féle szerepet szeretnék alkalmazni: felhasználó, feltöltő, adminisztrátor. A felhasználó az, aki játszani tud az oldalon. A feltöltő tudja hozzáadni az újabb mérkőzések statisztikáit, de nem tudja módosítani a meglévőket. Az adminisztrátor jogot kap meglévő adatok törlésére, módosítására.

Oldalam játék része egy kártyás játékot foglal magában. Itt a játékosok különböző csomagok nyitásával bővíthetik csapatukat és játszhatnak mérkőzéseket. A csomagok kreditekbe kerülnek, amiket a játékosok a mérkőzéseken elért eredmények fényében kapnak meg. Kezdő csomag természetesen minden regisztrálónak jár, hogy biztosítva legyen számára a mérkőzések lejátszása.

Az projektem statisztikákkal foglalkozó része magában foglal többféle listázási lehetőséget, amik segítségével mindenki könnyen megtalálhatja a számára fontos játékost, csapatot. Emellett a legek között megtalálhatók azok, akik kimagaslóan teljesítenek az idény folyamán. Ez lényegében egy toplistát foglal magában.

Projektem megvalósítása során több programozási nyelvet is szeretnék használni: PHP, JavaScript, HTML, CSS, Bootstrap. Ezek segítségével fogom megvalósítani a frontend és backend részét weboldalamnak.

A feladatok jelentős részében a PHP programozási nyelv adta lehetőségeket szeretném kihasználni, de fontos lesz a JavaScript is, mivel szeretném interaktívvá tenni a játék felületét.

Felmerülhet a kérdés, miért csinál valaki webes játékot? Tapasztalatom szerint napjainkban az emberek nem szeretnek letölteni dolgokat, könnyebb és gyorsabb számukra egyszerűen csak beírni egy címet a böngészőbe és elindítani a játékot.

Azért ezt a témát választottam, mert érdekel a kosárlabda világa és az ehhez fűződő dolgok. Szeretnék egy olyan felületet biztosítani rajongó társaimnak, ahol informálódhatnak és játszhatnak is egyben. Számomra nagyon fontos a projektem jó megvalósítása, mivel szeretném a későbbiekben elindítani egy működő weboldal formájában a projektemet, hogy valóban együtt játszhassak azokkal, akiket hasonló módon érdekel a kosárlabda világa.

# 2. A témaválasztás indoklása

Ebben a részben szeretném ismertetni, miért döntöttem a kosárlabda játék készítése mellett. Nagyon sokat gondolkodtam, mi lehetne megfelelő téma, ami érdekes számomra, emellett kihívást jelent megvalósítani.

A kosárlabda játék alap ötlete a FIFA nevezetű focis játékból ered. Ennek a játéknak van egy olyan része, amikor a felhasználó összegyűjtheti a focisták kártyáit, saját csapatot csinálhat belőlük, majd mérkőzéseket vívhat csapatával.

Ehhez hasonló kosárlabdás játékot szeretnék elkészíteni. Kerestem hasonló webes alapú játékot, sajnos nem találtam olyat, ahol mérkőzéseket is lehet játszani az összegyűjtött csapatokkal. Ezt szeretném megvalósítani magyar nyelvű felülettel, mert az általam talált kártyás játékok többnyire angol nyelvűek. Tapasztalataim szerint egyre több magyar anyanyelvű személy érdeklődik a kosárlabda iránt, ezért szeretnék számukra is könnyen játszható felületet biztosítani programommal.

A hasonló jellegű oldalakon felleltem több olyan opciót, melyeket saját projektemben szeretnék másként megvalósítani. Példaként a legtöbb ilyen játékban egy játékosnak több verziója is megszerezhető, ellenben az én játékomban minden játékos csak egy verzióban lesz fellelhető.

Mindig is érdekelt a sport világa, úgy érzem, ezzel a feladattal még jobban beleláthatok a kosárlabda érdekességeibe. Szeretném, ha mások is hasonlóképpen tekintenének erre a gyors és rendkívül izgalmas sportra, mint én. Ez a projekt segítségére lesz mindenkinek, hogy jobban megértse a kosárlabda világát és meg is szeresse azt.

Természetesen lényeges számomra, hogy ne csak érdekes legyen ez a feladat, hanem a szakmai fejlődésemet is segítse. Ezért szeretném a frontend részt látványosan megvalósítani, emellett a játék menetének gördülékenységét is hasonlóan folyamatosra csinálni, mint amilyen maga a kosárlabda játéka a való életben.

Gyerekkoromban nagyon szerettem a kártyás játékokat, ezért gondoltam arra, hogy jó lenne az alkalmazásomat kártyás formában megvalósítani. Napjainkban a legtöbb ember online játszik, nem fizikai formában, viszont a kártyázás élménye ilyen formában is megvan.

Az általam készített játék nagy élményt nyújt minden embernek, aki szeret gyűjteni, mivel saját csapatot kell gyűjteni és fejleszteni. Emellett a szerencsejáték is része lesz, természetesen csak kitalált valutával. Különböző csomagokat lehet kinyitni a játékban, melyek játékosokat, vagy valutát tartalmaznak.

Összefoglalva szeretnék egy olyan felületet biztosítani a magyar kosárlabda rajongóknak, akik kikapcsolódnának egy ilyen jellegű játékkal és szeretném vonzóvá tenni ezt a nagyszerű sportot mindenki számára. Ezzel a céllal készítem el a vizsgamunkámat.

# 3. Fejlesztői dokumentáció

## 3.1. Az alkalmazott fejlesztői eszközök

Itt szeretném ismertetni a projektem megvalósítása során használt fejlesztői eszközöket. Főként a PHP programozási nyelvet használtam a feladat megvalósítása során. Emellett az alkalmazás egyes részeinél a JavaScript nyelvet is felhasználtam. Főként a felhasználó számára való visszajelzésre használtam fel ezt a technológiát.

Az adatbázis elkészítését és a hozzá kapcsolódó műveleteket a phpMyAdmin és MySQL segítségével valósítottam meg. A frontend részt CSS és Bootstrap segítségével készítettem el.

A vázrajzok készítéséhez használtam a canva.com nevű online tervező alkalmazást. Ennek segítségével könnyen el tudtam készíteni a tervezett felületeket. A képek és ábrák szerkesztéséhez az Adobe Photoshop programot használtam.

A programozási műveletek és a megjelenés megírását a Brackets nevű programmal végeztem. Ez nagyban meggyorsította a munkámat, főleg a design elkészítésénél volt hasznos számomra.

A dokumentáció elkészítése során a Microsoft Office csomag Word programját használtam. Ezáltal könnyen meg tudtam valósítani a dokumentáció igényes megformázását.

## 3.2. Felületterv

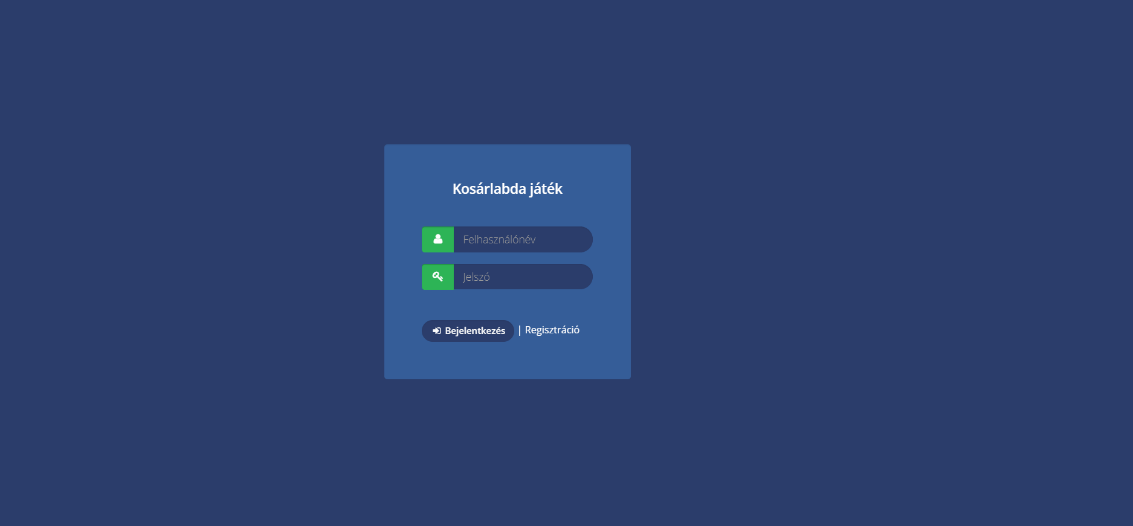
Ebben a részben szeretném ismertetni, hogyan képzelem el a projektem kinézetét és logikáját. Ismertetni fogom az alkalmazásommal kapcsolatos főbb tudnivalókat, melyek segítségével látható lesz egy szakmabeli személy számára a program működése, ezzel segítve a továbbfejlesztését.

Az alkalmazásomban elérhető funkciók nagymértékben függnek attól, hogy a bejelentkezett felhasználó milyen jogosultságokkal rendelkezik. A későbbiekben ismertetem a funkciókat részletesen is, most csak a főbb elemeket említem meg.

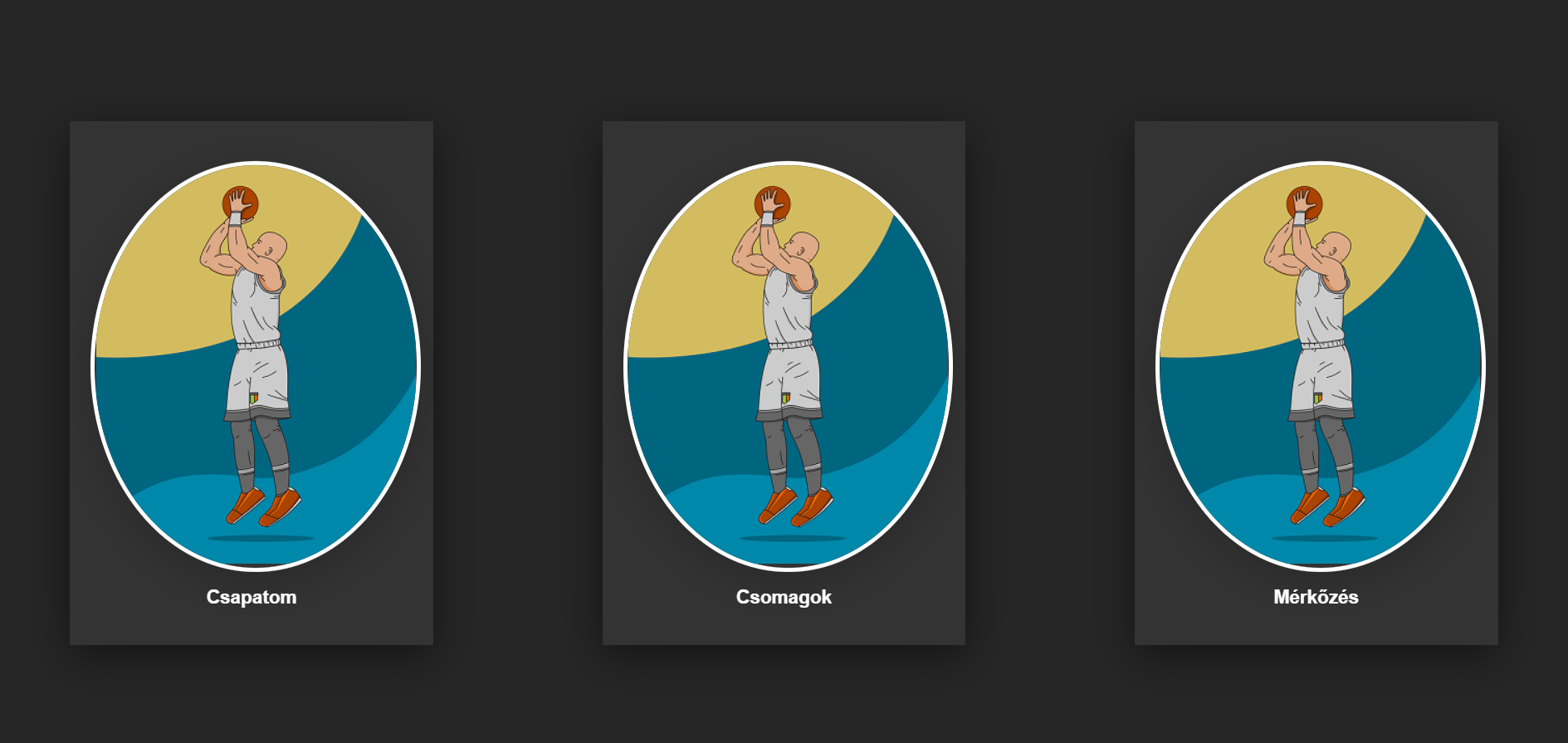
Regisztráció után az alapértelemzett jogosultság a felhasználó. Egy másik jogosultság a moderátor. Aki ezzel rendelkezik számára elérhető az a lehetőség, hogy tudja változtatni a játékosok adatait és az ellenfelek táblában lévő csapatok adatait. Emellett egy admin jogosultság lesz még programomban, akinek módjában áll változtatni a felhasználók jogosultságát és a csapatokkal kapcsolatos adatokat szerkeszteni.

Az alkalmazásom indítóoldala a bejelentkezési felület lesz. Ezen semmi extra funkció nem lesz megtalálható, két beviteli mező és két gomb. Az egyik gomb a bejelentkezési gomb lesz, míg a másik át fog irányítani a regisztrációs oldalra. Emellett egy elfelejtett jelszó funkció is helyet kap az oldalon. A bejelentkezéshez szükséges adatok a felhasználónév és a jelszó, melyek megadása után a felhasználó elkezdheti a játékot. Az elfelejtett jelszó gomb szintén egy oldalra irányít át, ahol a felhasználó segítséget kap, ha elfelejtette jelszavát és szeretne újat kérni.

A regisztrációs oldal hasonló, mint a bejelentkezési felület, viszont több beviteli mezőt fog tartalmazni. Szükséges lesz megadni a felhasználónevet, egy email címet és egy jelszót. A regisztráció megtörténte után megjelenik egy felugró panel, amely megköszöni a regisztrációt, majd a panelen látható gomb megnyomásával átirányítás történik a bejelentkezési felületre. Itt látható a bejelentkezési oldal tervezett kinézete.



Ahogy említettem a bejelentkezés megtörténte után elindul a játék. Először a felhasználó jogosultsággal rendelkezők szempontjából szeretném leírni a játék menetét. Azonnal megjelenik egy választó menü, ahol a felhasználó eldöntheti, hogy megnézi a csapatát, csomagot nyit, vagy mérkőzést játszik. Ezt az alábbihoz hasonló formában tervezem megvalósítani.

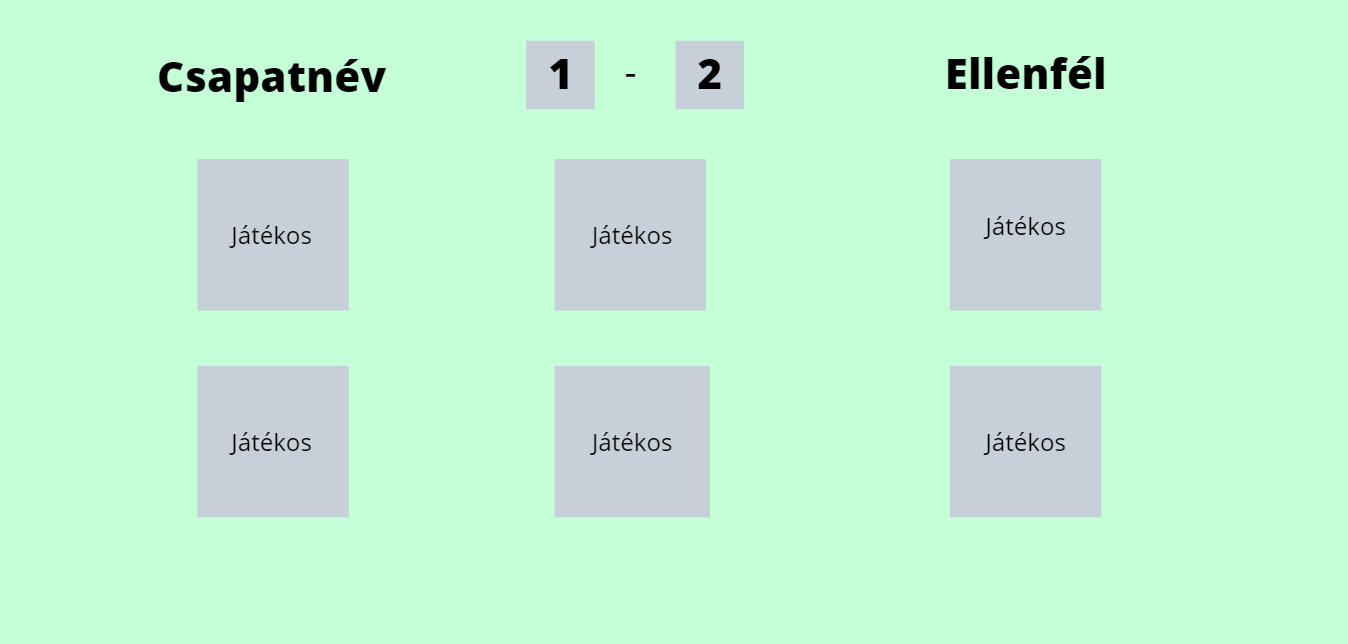


Ezen a felületen emellett még meg fog jelenni a felhasználó eddigi statisztikája. Itt megtekintheti eddig mennyi mérkőzést vívott, ebből mennyit nyert meg, mennyit veszített el.

Amennyiben a csapata megtekintését választja, megjelennek táblázatos formában a játékosai és lehetősége lesz megtekinteni az adataikat, vagy eladni őket. Ez a menüpont arra szolgál, hogy láthassa a felhasználó, milyen erősségűek a játékosai és együtt tekinthesse át őket.

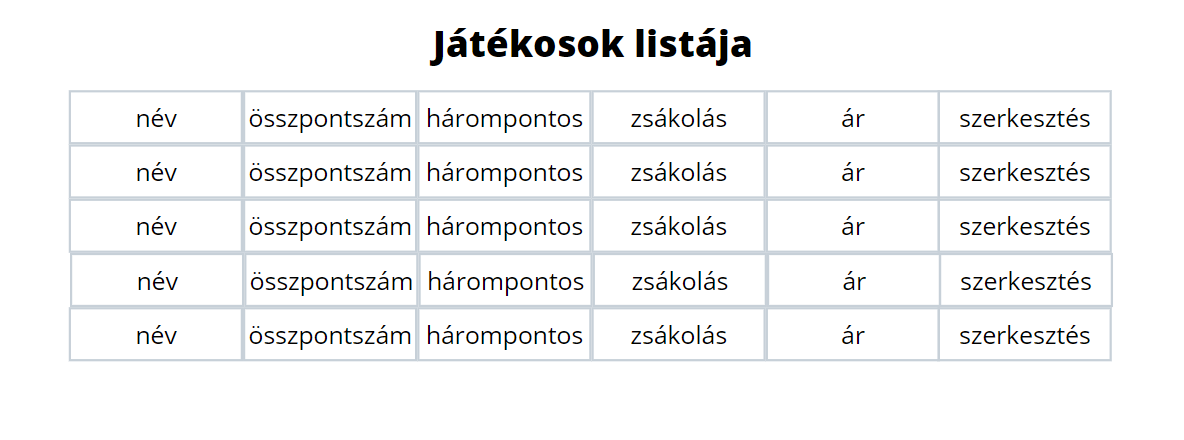
A csomagok menüpont hasonlóan fog kinézni, mint a menü rész. Több lehetőség közül választhatja ki, mekkora értékű csomagot szeretne venni. A csomagokat ezután ki is nyithatja, majd eldöntheti, hogy csapatához adja-e a játékost, vagy eladja a játékos értékének megfelelő összegért.

A mérkőzés menüpont mérkőzések lejátszására szolgál. Eldöntheti a felhasználó, hogy milyen erősségű ellenféllel szemben szeretné lejátszani a mérkőzést, majd elkezdődhet a játék. Itt meg lesz jelenítve a két csapat neve és az aktuális eredmény. Ezalatt látható lesz a játékosok kártyáinak listája.



Amikor a felhasználó rákattint egy játékos kártyájára lehetősége nyílik kiválasztani, hogy a játékos mely tulajdonságával szeretne párbajt vívni az ellenfél játékosával. A párbaj úgy zajlik, hogy mindkét csapat esetén ki kell választani egy játékost, majd a választott tulajdonság értékének összehasonlítása történik. A győztes játékos szerez a csapatának egy pontot. Ezután újabb párbaj következik. A mérkőzések addig tartanak, míg egyik csapat el nem éri a 10 pontot. A mérkőzések után visszatér a menübe.

Amennyiben moderátor jogosultsággal történik a bejelentkezés a menüben más opciók jelennek meg. A menü felülete hasonlóképpen jelenik meg, mint a már korábban ismertetett menü. A megjelenő opciók a következők: ellenfél csapatok szerkesztése és játékosok szerkesztése. A szerkesztés mellett új elemek hozzáadására is lehetősége lesz a moderátornak. Az ellenfél csapatok szerkesztésénél változtatásokat tehet a csapat nevében, a csapatban szereplő játékosokat is megváltoztathatja és a csapat szintjét is megváltoztathatja. Lényegében a moderátori fiók a felhasználókkal kapcsolatos módosításokat nem tudja megtenni, viszont a játékban szereplő elemekkel mindenféle változtatást meg tud tenni. A játékosok listázása ehhez a vázrajzhoz hasonló felülettel lesz megvalósítva.



Az admin jogosultsággal történő bejelentkezés után az előbbiekben már ismertetetthez hasonló menüpont fog megjelenni, viszont itt az opciók a felhasználókkal kapcsolatosak lesznek. Választhat a felhasználók szerkesztése és a csapatok szerkesztése opciók között. A felhasználók szerkesztése menüpontnál lehetősége lesz módosítani a felhasználók felhasználónevét, a jogosultságukat, a pénzüket és lehetőségük lesz aktiválni, vagy inaktiválni az adott felhasználót. Mindezek mellett a felhasználóhoz tartozó csapatot is lehetősége lesz módosítani. Táblázatos formában lesz erre is lehetősége.

A csapatokban lévő játékosok maximális létszáma 12 fő. Amennyiben a csomagok nyitásánál átlépné az illető a létszámhatárt lehetősége lesz megkapni a játékos árának megfelelő összeget a pénzéhez.

Regisztráció után a felhasználó kap egy öt darab kártyát és tízezer pénzösszeget a vagyonához. Azért kap öt játékost, mert a mérkőzés elindításához legalább ennyi kártyára van szüksége. Kevesebb kártyával nem lesz lehetősége mérkőzést játszani. A kezdeti vagyon azért lényeges, hogy már induláskor tudjon csomagokat nyitni a játékos.

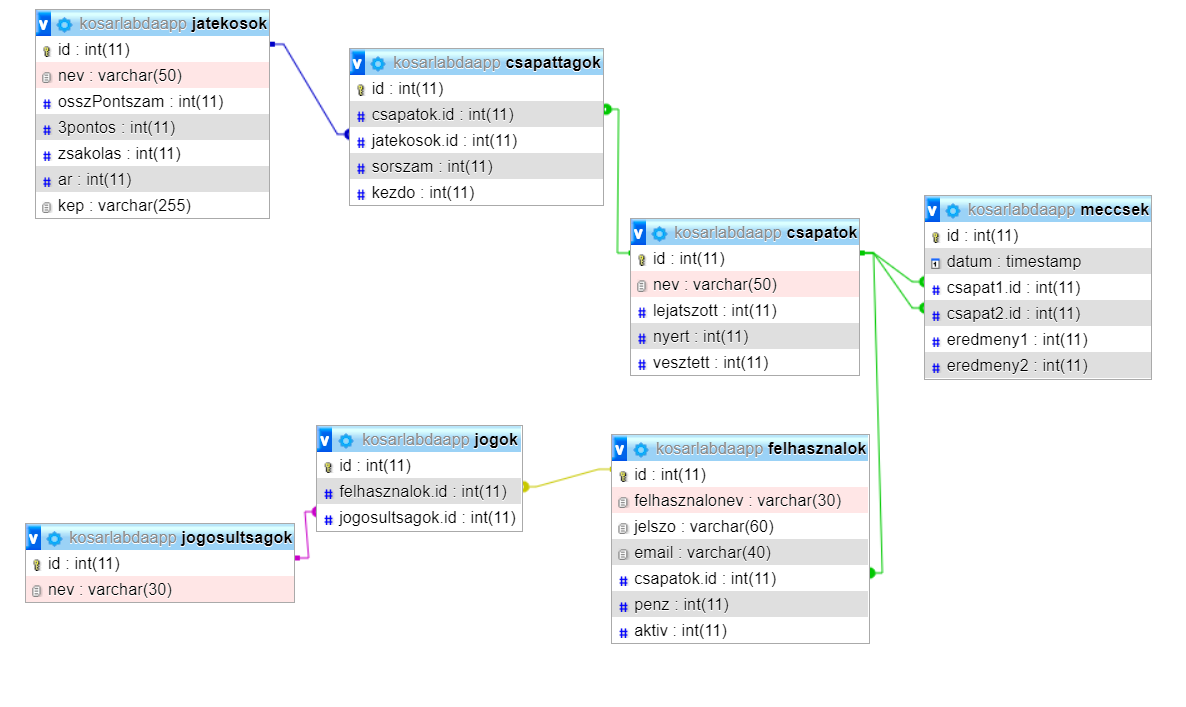
A csomagok között többféle opció is megjelenik. Elérhető lesz 2000-10000 értékű csomag. Ezen csomagok kinyitásával a felhasználók bővíthetik csapataik létszámát. Ezek a csomagok vagy játékost fognak tartalmazni, vagy egy pénzösszeget, mely a csomag áránál kevesebb is lehet. Ezzel szeretném élvezetessé tenni a játékot. Nem minden esetben nyer ezáltal a felhasználó, esetenként rosszul is kijöhet ebből.

A felhasználók pénzösszegüket nem tudják feltölteni, csak mérkőzések lejátszása után kapnak a mérkőzés kimenetelének megfelelő összeget, vagy csomagok nyitása útján szerezhetnek pénzt.

A felhasználói fiókok törlésére nem lesz lehetőség. Az adatbázisomban elérhető lesz aktív néven egy mező, mely arra szolgál, hogy mutassa a felhasználó aktív-e, vagy sem. Ez azért lényeges, mert az adatok felbecsülhetetlen értékűek, ezért nem helyénvaló törölni a meglévő adatokat. Amennyiben a fiók inaktív nem lesz lehetőség a bejelentkezésre és az ehhez kapcsolódó folyamatokra, mint a saját csapat menedzselése és mérkőzések lejátszása.

A fentebb ismertetett tervek alapján fogom megvalósítani a projektemet, mely remélem valóban hasznos applikáció lesz a téma iránt érdeklődőknek.

## 3.3. Adatmodell leírása

A programhoz tartozik egy adatbázis is, melyet a fentebbi képen láthat. Az adatbázis kosarlabdaapp néven lett elnevezve és tartalmaz 7 táblát. A 4 törzsadatok tárolására szolgáló tábla a következő: felhasznalok, csapatok, jatekosok, jogosultsagok. A 2 tranzakciós tábla jogok és csapattagok névre hallgat. Az adatbázisom tartalmaz egy naplózásra szolgáló táblát is, mely a mérkőzések eredményeinek eltárolására szolgál. Ennek neve meccsek.

Minden táblában megtalálható egy id nevű mező, mely a rekordok azonosítására szolgál. Ez a mező elsődleges kulcs és auto increment beállításokat kapott.

A felhasznalok tábla a regisztrált felhasználók adatainak eltárolására szolgál. A táblában eltároljuk a felhasználónevet, az email címét, a jelszavát titkosított formában, a hozzá tartozó csapat azonosítóját, a felhasználó pénzét. Az aktiv nevű mező a felhasználó aktív, vagy inaktív státuszának eltárolására szolgál. Az aktív státuszt 1-es, az inaktív státuszt 0-ás értékkel jelöljük. A csapatok.id mező egy idegen kulcs is, mely a csapatok tábla id mezőjére mutat. A táblában lévő id mező szintén idegen kulcs a jogok tábla felhasznalok.id mezőjére mutatva.

A csapatok tábla a felhasználók csapatainak eltárolására szolgál. Itt tároljuk a csapat nevét, az lejátszott, a megnyert és az elvesztett mérkőzések számát is. A táblában lévő id mező idegen kulcsként funkcionál, a csapattagok táblában lévő csapatok.id mezőre mutat.

A jatekosok táblában találhatók a játékban szereplő kosárlabdázók adatai. Itt tároljuk a kosárlabdázók nevét, 3 féle pontszámát (osszPontszam,3pontos,zsakolas), árát és a hozzá tartozó képet. A képek tárolását a kepek\jatekosok mappában valósítottam meg. Az itt található id mező egy idegen kulcs, mely a csapattagok táblában lévő jatekosok.id mezőre mutat.

A jogosultsagok tábla a lehetséges jogosultságokat tartalmazza, ezek a következők: admin, moderator, felhasznalo. Az itt található id mező egy idegen kulcs, mely a jogok táblában lévő jogosultsagok.id mezőre mutat.

A jogok táblában van eltárolva, hogy melyik felhasználó milyen jogosultságokat kapott. A táblában az id mezőn kívül csak a felhasználó és a jogosultság azonosítóját találhatjuk.

A csapattagok tábla annak meghatározására szolgál, hogy az adott csapatot mely játékosok alkotják. A tábla tartalmaz egy csapat és egy játékos azonosító mezőt. Emellett megtalálható itt egy sorszam mező, ez a játékos csapatban elfoglalt sorszámát tárolja és egy kezdo mező, mely értéke 1, ha az adott kosárlabdázó a kezdőcsapat tagja, illetve 0, ha nem kezdő a kosárlabdázó.

A meccsek tábla a mérkőzések végeredményének eltárolására szolgál. Eltároljuk a mérkőzés dátumát, a két csapatot és mindkét fél eredményét. A csapat1.id és a csapat2.id is idegen kulcsok, melyek a csapatok tábla id mezőjére mutatnak.

Ezek voltak az adatbázishoz tartozó leírások, az itt található részletes bemutatás segítségével könnyen megérthető a szerkezete.

## 3.4. Részletes feladatspecifikáció, algoritmusok

Ebben a fejezetben szeretném ismertetni a lényeges osztályait, függvényeit, metódusait. Ezáltal a program logikai menete könnyebben érthető lesz más fejlesztők számára is.

Minden oldalt a tagolva készítettem el, egy oldalhoz 3 rész tartozik. Az oldal alapszerkezetét adó php fájl, egy classes mappában található fájl és egy design mappában található fájl. Az oldal alapszerkezetét adó php fájlban történik a hozzá tartozó adatbázis műveletek meghívása is. A classes mappában található php fájlokban az oldalakhoz kapcsolódó adatbázis műveletek lettek eltárolva. A style mappában lévő css fájlok az adott oldalhoz kapcsolódó megjelenésért felelősek. Az összekapcsolódó részek könnyen megkülönböztethetőek, mivel az elnevezésük utal a tartalmukra. Például: csapatom.php, csapatomclasses.php, csapatomstyle.css.

A classes és design mappákban szerepel néhány fájl, melyekre több oldalon is szükség van, ezért kiszerveztem egy külön fájlba és meghívtam a szükséges helyeken. Ilyen a classes mappában szereplő adatbazisclasses.php és a jatekoskartyaclasses.php. A design mappában a kartyastyle.css és a jatekoskartyastyle.css fájlokat használtam több helyen is.

Az index névvel ellátott fájlok a kezdőoldalon található bejelentkezési felület megvalósításáért felelnek. A bejelentkezett állapotot a bejelentkezettAllapot session segítségével tárolom el. Az értéke bejelentkezett felhasználó esetén a hozzá tartozó id értéke, kijelentkezett felhasználó esetén egy üres string. Az indexclasses.php fájlban található sessionBejelentkezettE nevű metódus, mely megvizsgálja, hogy a felhasználó be van-e jelentkezve.

A regisztracio nevű fájlok a regisztráció működéséért felelősek. A jelszónak legalább 10 karakter hosszúságúnak kell lennie. Sikeres regisztráció után a bejelentkezési oldal nyílik meg.

A menu elnevezésű fájlok a bejelentkezés után látható főmenüt valósítják meg. A felület bal felső sarkában a felhasználó zsetonjainak száma, az oldal tetején középen a csapatának a neve, a jobb felső sarokban egy kijelentkezés gomb jelenik meg, mely visszaléptet a bejelentkezési oldalra és a bejelentkezettAllapot nevű session értékét üres stringre állítja. A mérkőzés menüpontra kattintva az ujEllenfel action hajtódik végre. Ez a rész készíti elő a későbbi ismertetésre kerülő merkozes fájlokat. Létrehoz több sessiont is, mégpedig: eredmeny1, eredmeny2, parbaj. Ezeket mind 0 kezdőértékkel inicializálja. Létrehozásra kerül egy sajatId, mely a felhasználó azonosítója és egy ellenfelId, mely az ellenfél azonosítója. Ezt do-while ciklus segítségével addig generáljuk, amíg nem talál egy a sajatId értékétől és a 7-es azonosítótól eltérő számot. Ezután készítettem két újabb sessiont sajatCsapat és ellenfelCsapat néven. Ezek tárolják a két csapat öt-öt kezdőjátékosát. Végül a merkozes.php oldalra történik átirányítás.

A csapatom nevű fájlok a saját csapat listázását és szerkesztését teszik lehetővé. A listázás két ciklusban történik, először a csapatban lévő kezdőjátékosok, majd a cserejátékosok kiírása történik. A játékosoknak megjelenik a neve, a képe, a hárompontos, az általános és a zsákolás pontszáma. Emellett a kezdő gomb segítségével kezdővé állíthatjuk, a csere gomb segítségével cserévé állíthatjuk az adott játékost. Az elad gomb lenyomásakor a játékos csapata a 7-es értéket vesz fel, így átkerül az eladottak csoportba.

A csomagok elnevezést kapó fájlok a csomagválasztási felületet valósítják meg. A felületen 3 kártya jelenik meg. A nyitás gombra kattintva a csomag értékének megfelelő értéket vesz fel a csomagAzon session. A legolcsóbb esetén 1, a köztes esetén 2, a legdrágább esetén 3. Emellett megvizsgálja a program, hogy van-e elég pénze a felhasználónak.

A csomagnyit névvel rendelkező fájlok a csomag kinyitását valósítják meg. Először a fizetes függvény levonja a csomag értékét a játékos pénzéből, majd a csomagTipus függvény random választ 0 és 1 között. 0 esetén pénzt, 1 esetén kosárlabdázót kap a felhasználó. Ezután a csomagAzon session vizsgálata történik. Ha 1-es értéket vesz fel a pénznyeremény 1 és 20000 közötti, a játékos általános pontszáma a legkisebb értékű játékos pontszáma és 85 közötti lehet. Amennyiben 2-es értéket vesz fel a pénznyeremény 1000 és 75000 közötti, a játékos általános pontszáma 75 és 85 közötti lehet. 3-as érték esetén a pénznyeremény 5000 és 200000 közötti, a játékos általános pontszáma 80 és a legnagyobb értékű játékos pontszáma közötti lehet. Az alsó és felső érték meghatározása a getMinOsszPontszam és a getMaxOsszPontszam függvények feladata. A randPontJatekosokSzama függvény megvizsgálja van-e az adott általános pontszámmal játékos az adatbázisban. Amennyiben van, a kosarasAzon függvény választ egy játékos a kellő általános pontszámúak közül. Ellenben a csomagAzon session 0-s értéket kap, vagyis pénznyereményt kap a felhasználó. Játékos nyitása esetén a csapattagE függvény megvizsgálja, hogy az adott kosárlabdázó tagja-e a felhasználó csapatának. Ha igen, akkor a csomagElad függvény a játékos árát adja a felhasználó pénzéhez. Ha nem tagja a csapatának, akkor a jatekosHozzaad függvény a csapattagok táblába helyez egy új értéket a felhasználó csapatának és a kosárlabdázó azonosítójával.

A merkozes elnevezésű fájlok a játékmenet megvalósítását tartalmazzák. Elsőként létrehozásra kerül egy vegeredmeny nevű session 0 kezdőértékkel. A felületen megjelenik a felhasználó csapatának 5 kezdőjátékosa kártyás kinézetben. Mindegyik kártya alján található egy párbaj gomb. A gomb lenyomása meghívódik a btnParbaj nevű action, ahol megtörténik a két párbajozó játékos kiválasztása a sajatCsapat és az ellenfelCsapat session értékeiből, majd a parbaj függvény meghívása. Ez a függvény összehasonlítja a két játékos 3 pontszámának összegét, majd parbajEredmeny session néven elmenti az eredményt szöveges formában és a megfelelő eredmeny sessionnek ad egy további pontot. Döntetlen eredmény esetén mindkét eredmeny session kap egy-egy pontot. Ezután átirányít a parbaj.php oldalra. A párbajban már részt vett játékosok helyére -1-es érték kerül a sajatCsapat és az ellenfelCsapat esetén is. Ezen érték esetén a listázáskor az adott helyen szereplő elem nem kerül listázásra. A párbajra alkalmas játékosok azok, akiknek az azonosítója szerepel a sessionben. Addig tart a mérkőzés, amíg a parbaj session nem éri el az 5-ös értéket. Ekkor a vegeredmeny session felvehet 1-es értéket győzelem esetén, 2-es értéket vereség esetén és 3-as értéket döntetlen esetén. Ekkor a parbaj session értéke 6 lesz, majd felugró ablakban értesíti a felhasználót a program az eredményről az eredmeny1 és eredmeny2 session értékét megjelenítve. Ha a parbaj session 6 és a vegeredmeny session nem nulla, a mérkőzés véget ért. Ebben az esetben egy kártya kerül a képernyőre, mely megjeleníti az eredményt, majd a vegeredmeny session 0 értéket kap, vagyis az eredeti értéke kerül visszaállításra. Amikor a parbaj session 6-os értéket vesz fel, megjelenik a jobb felső sarokban egy tovább gomb, mely a menübe történő visszalépést teszi lehetővé. Amikor a parbaj értéke 6 és a vegeredmeny értéke 0, akkor véget ért a mérkőzés, a menübe történik átirányítás. A mérkőzés végén a sajatGyozelem, ellenfelGyozelem és dontetlen függvények rögzítik az adatbázisban a csapatokhoz tartozó eredményt és a meccsek táblába a mérkőzés végső adatait.

A parbaj elnevezésű fájlok szorosan kapcsolódnak az előbb leírtakhoz. Ezen a felületen megjelenik a két párbajozó játékos és kiírásra kerülnek az adataik. A parbajclasses a jatekoskartyaclasses osztályból származik le, ahol a játékosok adatainak lekérdezése történik.

A fiokom elnevezésű fájlok a saját fiók adatainak szerkesztésére szolgáló felület megvalósítását tartalmazzák. A get szóval kezdődő nevű függvények a jelenlegi adatokat adják vissza. A Módosítás gomb lenyomására meghívódik az adatModositas függvény, mely a felhasználónév, az email cím és a csapatnév módosítását teszi lehetővé. A jelszó módosítás gomb lenyomására megjelenik két új mező, egyikbe a jelenlegi, a másikba az új jelszót kell írni, majd a jóváhagyás gombra kattintva a jelszoValtoztatas függvény végrehajtja ezt, amennyiben a jelenlegi jelszó valóban helyes. Az inaktiválás gombra kattintva a fiók inaktiválásra kerül, majd az index.php oldal nyílik meg. A felhasznalok táblában az aktiv mező értéke 0 lesz, ekkor a felhasználó nem fog tudni bejelentkezni.

A jatekosokszerk nevű fájlok a kosárlabdázók adatainak listázását tartalmazzák. Amennyiben átírunk egy adatot és a módosítás gombra nyomunk meghívódik a jatekosModositas függvény, mely segítségével megvalósul az adatok módosítása. A bal felső sarokban lévő felvétel gombra kattintva megnyílik egy modal, mely a felvételhez szükséges beviteli adatokat kéri. Minden adat megadása szükséges. Az adatok a felvétel sikerességéig lementődnek a felv névvel kezdődő sessionökben. Ezek kezdőértéke egy üres string.

A felhasznalokszerk nevű oldalon található felület hasonló megjelenésű, hasonló funkciókkal is rendelkezik. Bármelyik adat módosítása esetén a sor végén lévő módosítás gombra kattintva a felhasznaloModositas nevű függvény kerül meghívásra. Itt a név, az email, csapatnév, a pénz és a jogosultág kerülhet módosításra. A sor alatt helyet kapott egy aktivitást módosító gomb. Ennek értéke aktív fiók esetén inaktiválás, ellenben aktiválás. Erre kattintva a setAktivitas függvény fut le, aktiválás esetén 1-es értékre állítva a felhasznalok tábla aktiv mezőjét, inaktiválás esetén nullára.

Az adatbazisclasses osztály az adatbázishoz való kapcsolódást valósítja meg, a legtöbb class a projektben ebből származik le. A jatekoskartyaclasses a kosárlabdázók adatait jeleníti meg. Azok a classok, melyeknél kosárlabdázók jelennek meg ettől az osztálytól származnak le.

A projektem főbb algoritmusainak ismertetése ezennel véget ért. Ezen rész a projekt logikai menetének megértésére szolgál, mely segítségével a továbbfejlesztés gördülékenyebben tud haladni.

## 3.5. Tesztelési dokumentáció

Ebben a fejezetben a tesztelés folyamatát szeretném bemutatni. A fejlesztés során végeztem egységteszteket, modulteszteket, integrációs teszteket, majd végül rendszerteszteket is. Ezek segítségével javítani tudtam az esetenként jelentkező hibákat, melyek a program helyes működését befolyásolták.

A tesztelésben során igyekeztem alkalmazni a fekete doboz és fehér doboz módszereket is, így minél ebbe a részbe bevontam más fejlesztőket is, akik némelyike ismerte a teljes kódot is. Emellett a tesztelés végső szakaszaiban szerepet vállaltak olyan személyek, akik nem ebben a szakmában tevékenykednek.

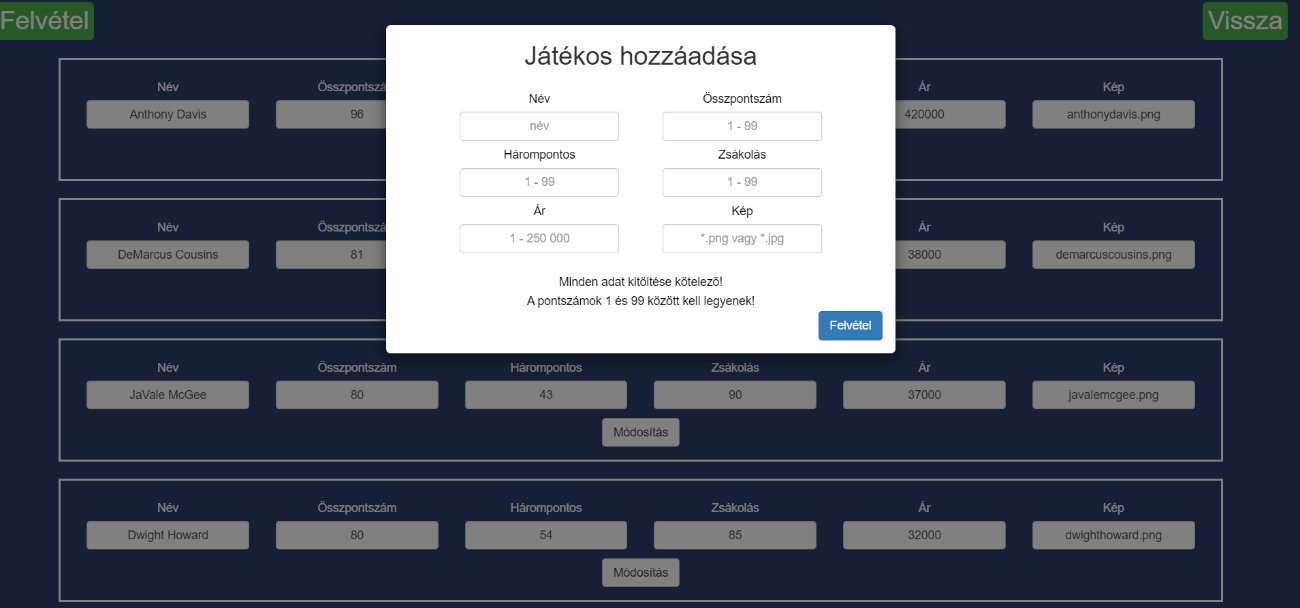
A legtöbb teszt a csomagok nyitására és a mérkőzésre vonatkozóan történt. Ezen tesztek során derült fény arra a hibára, hogy ha a felhasználó kinyitott egy csomagot, majd a tovább gomb helyett frissített az oldalon új csomagot kapott, viszont csak az első után történt fizetés. Ezt a hibát úgy javítottam, hogy a valós fizetés a csomagok.php oldalon található nyitás gombra kattintás helyett a csomagnyit.php oldalon történik, mikor megjelenik egy új kártya.

A tesztelés újabb szakasza során derült fény a saját csapat listázását megvalósító oldalon található hibára. Egy játékos eladása esetén több játékos is eltűnt a csapatból. Ezt a hibát a listázás nem helyes módon történő megvalósítása okozta, melyet kijavítottam, így már rendeltetésszerűen működik a programom ezen része is.

A mérkőzés menüpont sok szempontból történő tesztelést igényelt. A tesztek rámutattak a rész javítandó eseteire. Ezáltal ki lett szűrve egy mérkőzés végeredményének többszöri elmentésének lehetősége az adatbázisba, egy kosárlabdázó többszöri párbajra kerülése egy mérkőzés során és az ellenfelek kiválasztásának folyamata is javításra került.

Az egész program fejlesztése során figyeltem a felhasználók helyénvaló tájékoztatására, ezzel is elejét véve a hibás adatok megadásának. Nem megfelelő adat beírása esetén felugró ablakok segítségével kap tájékoztatást a felhasználó, hogy mit kell változtatnia, milyen értékre.

A kosárlabdázók szerkesztésére szolgáló oldalon található felvétel esetén a beírt adatok hibás megadásakor az adatokat elmentettem és eltároltam, hogy a felhasználónak a már helyesen beírt adatokat ne kelljen újra kitöltenie, hanem csak a rosszul megadott adat módosítását kelljen megtennie.



A beviteli mezők esetén törekedtem a jól kezelt adatbevitelre, ezáltal is megakadályozva a program rendellenes működését.

Amennyiben az adatbázis konfigurálása nem jól történik, a program működése nem elvárható, mivel a legtöbb művelethez szükséges az adatok lekérése és új adatok rögzítése az adatbázisba.

Egy böngészőben futó program esetén nagy jelentősége van a több böngészőben is történő tesztelésnek. Én, mint korábban is említettem a fejlesztés során a Google Chrome felületét használtam. Emellett a Microsoft Edge, a Google Chrome mobil változata és a Samsung Internet mobil változata volt használva a tesztelés során.

A tesztelés folyamatát nagyon hasznosnak találtam, mert több javításra szoruló részre fény derült ezek során, amiket a fejlesztés újabb szakaszában javítani tudtam.

## 3.6. Továbbfejlesztési lehetőségek

A most következő rész a dokumentáció egyik legérdekesebb fejezete, mivel egy programról sosem mondható el, hogy teljesen elkészült. Mindig vannak olyan területek, amelyek fejleszthetőek, jobbá tehetőek még. Most szeretném ismertetni, hogy milyen területeken fejleszteném még tovább a programom.

Elsőként az elfelejtett jelszó részt lehetne fejleszteni, jelenleg egy felugró ablakban megjelenik egy email cím, ahol a fejlesztőkkel lehet felvenni a kapcsolatot. Ezt úgy lehetne megvalósítani, hogy megjelenik egy felület, ahol a felhasználónév és email cím megadása után meg lehet változtatni a jelszót.

A mérkőzés menetének átalakítása is megvalósítható a jövőben. Úgy tenném izgalmasabbá a játékot, hogy a felhasználó nem csak a kosárlabdázót választhatja ki, hanem annak pontszámát is, amit szeretne a párbaj során összehasonlítani.

A legnagyobb módosítás a felhasználók és kosárlabdázók szerkesztése oldalakon történne. Mindkét oldal tetejére egy kereső sávot illesztenék. Az ebbe beírt szöveg alapján történne a listázás, ezáltal a szerkeszteni kívánt sor adatai könnyen megtalálhatóak.

A jövőben szeretném ezeket a fejlesztéseket megvalósítani az alkalmazásomban, ezáltal még jobbá tenni a programot.

# 4. Felhasználói dokumentáció

## 4.1. A program általános specifikációja

Ebben a fejezetben szeretném a leendő felhasználók számára ismertetni az általam elkészített alkalmazás főbb jellemzőit és funkcióit.

A program készítése előtt és alatt többször végiggondoltam miért lesz hasznos a munkám. Úgy látom napjaink gyorsan haladó világában az emberek egyre kevesebb időt tudnak fordítani arra, hogy konzol, vagy számítógép elé üljenek játszani. A telefon viszont minden embernél rendszeresen ott van. Ezért tartom hasznosnak a programot, melyet készítettem. Kötetlenséget nyújt, mivel nem szükséges nagy kijelző a használatához. A fejlesztés során komoly figyelmet fordítottam arra, hogy teljes körűen reszponzív felületet valósítsak meg, mely minden méretű kijelzőn egyformán élvezetes megjelenést biztosít.

Napjainkban egyre több ember kedveli a kosárlabda világát, mivel egy rendkívül gyors játék, melyben hirtelen változhat a mérkőzés aktuális állása. A fejlesztés során törekedtem egy ehhez hasonló játékmenetet kialakítani. A játékos akár egy perc alatt és végezhet egy mérkőzéssel.

A játékmenet elsajátítása nem igényel komolyabb előzetes ismeretet, ezért azok is jó lehetőséget láthatnak a program használatában, akik nem érdeklődnek a kosárlabda világa iránt. A program használata kinyit ezeknek a személyeknek számára egy új világot, mivel a kosárlabdázók kártyái ismertetik a játékosok erősségeit és gyengeségeit.

A program megvalósítása jó lehetőséget kínál a kártyajátékokat szerető személyek számára is. Mivel a kártyákon lévő pontszámok egyértelműen mutatják, melyik kosárlabdázó jobb a játékmenet szempontjából, így nem szükséges, hogy a felhasználó ismerje az összes kártyán szereplőt.

Az előkészületek során egy élvezetes alkalmazás elkészítését terveztem, mely jó lehetőség minden korosztály számára. Emellett az is célom volt a fejlesztés során, hogy azok is könnyen használhassák az alkalmazást, akik nem jártasak a sport világában, anélkül, hogy bármilyen hátrány érné őket.

Egy egyszerű felületet és logikus gondolkodással könnyen használható alkalmazást szerettem volna elkészíteni. Úgy érzem sikerült megvalósítanom ezt, ezáltal bárki élvezheti az elkészített játékot.

## 4.2. Rendszerkövetelmények

Ebben a fejezetben szeretném ismertetni az alkalmazásom futtatásához szükséges követelményeket.

Az alkalmazásom futtatásához egy böngésző, egy webszerver és egy adatbázisszerver szükséges. A további fejezetekben az általam használt eszközöket fogom ismertetni, melyek a következők: Google Chrome böngésző és XAMPP webszerver-csomag.

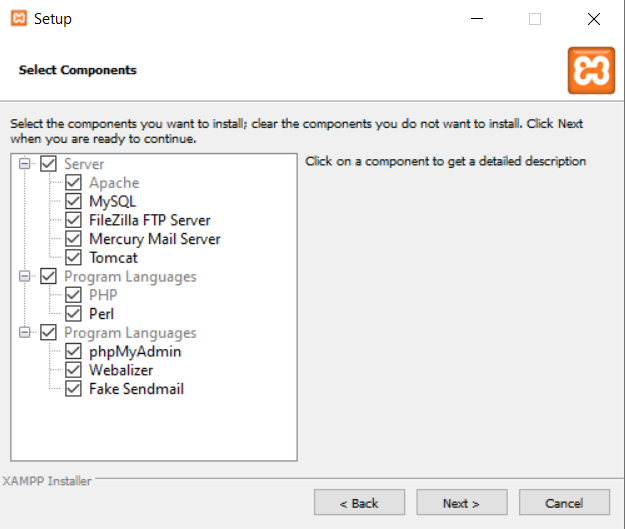
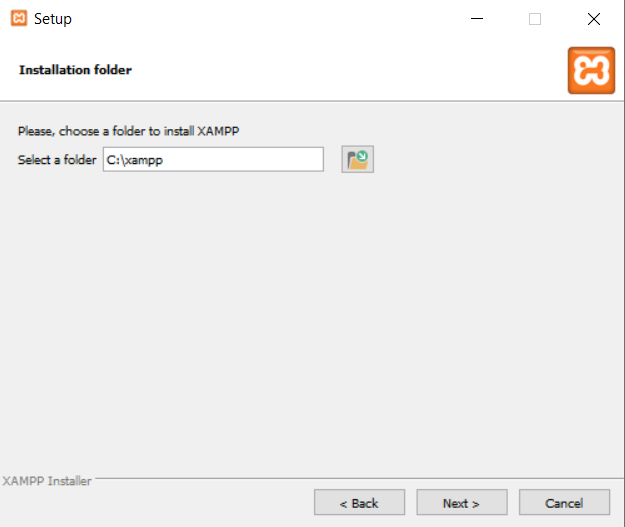
Operációs rendszert illetően Windows, Linux, Mac, Android és iOS rendszereken egyaránt alkalmas az alkalmazás használatára.

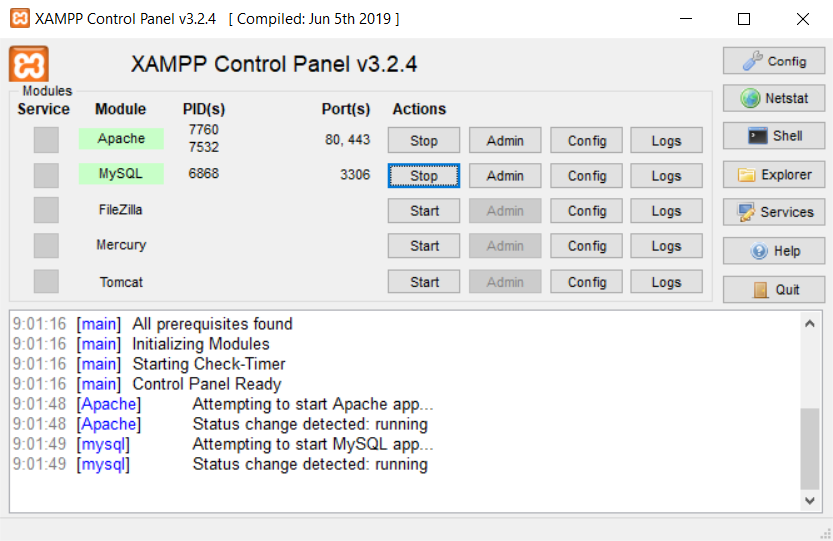
A hardverigények terén Windows és Linux operációs rendszert futtató eszközön Intel Pentium 4, vagy újabb, SSE2 utasításkészletet használó processzor szükséges. A Google Chrome futtatásához a Windows 7, vagy ennél újabb operációs rendszer szükséges. Amennyiben Android operációs rendszert futtató eszközt használ Android 4.4, vagy újabb rendszer szükséges.

## 4.3. A program telepítésének és konfigurálásának a leírása

A dokumentáció ezen része a telepítéshez nyújt segítséget. Részletesen leírom az alkalmazásom futtatásához szükséges programok telepítését, az adatbázis konfigurálását és a program futtatásához szükséges fájlok helyes elhelyezését. Ezeket a lépéseket Windows operációs rendszert futtató számítógépen mutatom be.

Ahhoz, hogy használni lehessen az általam készített programot az előző fejezetben ismertetett szoftverekre lesz szükség. Emellett az adatbázis konfigurálását is meg kell tenni. Ezután az alkalmazásom futtatásához minden rendelkezésre áll.

Első lépésként az XAMPP szoftvercsomag telepítése szükséges. Az XAMPP 7.4.3-as verzióját fogom használni. Miután elindítottuk a telepítő fájlt, engedélyt kér a, hogy módosításokat hajthasson végre az eszközön. Az igen gombra kell kattintani. Ezután megjelenik az XAMPP telepítő üdvözlő része, a next gombra kell kattintani. Ekkor megjelenik egy felület, melyen kiválaszthatjuk, hogy a szoftvercsomag mely elemeit szeretnénk telepíteni. Hagyjuk bepipálva az összes négyzetet, majd kattintsunk a next gombra. Ezután a program telepítésének helyét kell megadnunk. Alapértelmezetten a C:\xampp mappába telepíti, ezt javasolt így hagyni. Ezután a next gombbal lépjünk tovább egészen addig, amíg a telepítés meg nem történik. Amikor végzett a program a telepítéssel nyomjunk a finish gombra. Ekkor az alkalmazás vezérlőpanel része elindul, miután kiválasztottuk a nyelvet és a save gombra kattintottunk.

A vezérlőpanelen el kell indítanunk az Apache és MySQL modulokat. Ezt a mellettük lévő start gombbal tehetjük meg. Az elindításuk sikerességét visszajelzi, hogy a start gomb felirata stop szövegre változik és a modulok neve zöld háttérszínt kap.

Ezt követően a MySQL mellett lévő admin gombra kell kattintanunk. Ez megnyitja a phpMyAdmin felületét. Itt a felső sávban lévő importálás fülre kell kattintanunk. Itt a fájl kiválasztása gombra kattintva válasszuk ki a kosarlabdaapp.sql nevű fájlt, mely az adatbázist tartalmazza. Ezután az oldal alján, a jobb alsó sarokban lévő indítás gombra kell kattintani az adatbázis konfigurálásához.

A Kosarlabdaapp nevű mappát el kell helyeznünk a következő helyen: „C:\xampp\htdocs”.

Ezután a program indításához a böngésző címsorába kell írnunk a következő szöveget: „localhost/Kosarlabdaapp/index.php”. Ekkor elindul a program.

## 4.4. A program használatának a részletes leírása

# 5. Összegzés