

Szakképesítés megnevezése: **Szoftverfejlesztő**

OKJ száma: **54 213 05**

ZÁRÓDOLGOZAT

Kosárlabda kártyajáték

Basket Card Manager

Györgyi Tamás

témavezető

Vorák Gergő

2/14.S

Budapest, 2020

Nyilatkozat

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

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

Dátum:

hallgató aláírása

Tartalomjegyzék

Nyilatkozat.....	2
Tartalomjegyzék.....	3
1. Bevezetés	4
2. A témaválasztás indoklása	6
3. Fejlesztői dokumentáció	8
3.1. Az alkalmazott fejlesztői eszközök	8
3.2. Felületterv	9
3.3. Adatmodell leírása	13
3.4. Részletes feladat-specifikáció, algoritmusok	15
3.5. Tesztelési dokumentáció	19
3.6. Továbbfejlesztési lehetőségek.....	21
4. Felhasználói dokumentáció.....	22
4.1. A program általános specifikációja	22
4.2. Rendszerkövetelmények	24
4.3. A program telepítésének és konfigurálásának a leírása	25
4.4. A program használatának a részletes leírása	27
5. Összegzés.....	31
Forrásjegyzék.....	33
Ábrajegyzék	33

1. Bevezetés

A szoftverfejlesztő OKJ-s képzés elvégzéséhez szükséges elkészítendő szakdolgozatom tervét szeretném ismertetni. A szakdolgozatomat webes formában fogom megvalósítani, témám az amerikai kosárlabda ligához (NBA) kapcsolódik. Egy kártyás játékot fogok megvalósítani, mivel kosárlabda témában kevés magyar nyelvű játék érhető el. Ezáltal a rajongók felfedezhetik a csapatokat és mérkőzéseket játszhatnak.

A kinézetre nagy hangsúlyt szeretnék fektetni a 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 fejlesztő szeretnék lenni, így nagyon fontos számomra a megfelelő jogosultságkezelés. 3 féle szerepet fogok alkalmazni: felhasználó, moderátor, adminisztrátor. A felhasználó az, aki játszani tud az oldalon. A moderátor tudja hozzáadni és módosítani a játékban szereplő kosárlabdázók adatait. Az adminisztrátor jogot kap a felhasználók adatainak módosítására is.

A látogatók különböző csomagok nyitásával bővíthetik csapatukat és játszhatnak mérkőzéseket. A csomagok zsetonokba kerülnek, amiket a játékosok a mérkőzéseken elért eredmények fényében kapnak meg. Kezdő csomag természetesen minden regisztrálónak jár, hogy biztosítva legyen számára a mérkőzések lejátszása.

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, hogy valóban együtt játsz hassak 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 alapötlete 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 programot, de 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 voltak. Tapasztalataim szerint egyre több magyar anyanyelvű személy érdeklődik a kosárlabda iránt, ezért az a célom, hogy számukra is könnyen játszható felületet biztosítsak a programommal.

A hasonló jellegű oldalakon találtam 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 elérhető.

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 segítségével 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 kialakítani és fejleszteni. Emellett a szerencsejáték is részese lesz, természetesen csak kitalált valutával. Különböző csomagokat lehet kinyitni a játékban, melyek játékosokat, vagy zsetonokat 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 elkészítése során. Emellett az alkalmazás egyes részeinél a JavaScript nyelv is szerepet kapott. Főként a felhasználó számára való visszajelzésre szolgált ez a technológia.

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

A vázrajzok készítéséhez használtam a canva.com nevű online tervező alkalmazást. Ezzel könnyen el tudtam készíteni a tervezett felületeket. A képek és ábrák szerkesztését az Adobe Photoshop programmal végeztem.

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

A dokumentáció elkészítése során a Microsoft Office csomag Word programját használtam. Ezáltal könnyen meg tudtam valósítani a dokumentáció igényes formázását. A PowerPoint program a prezentáció elkészítésénél volt segítségemre.

3.2. Felületterv

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

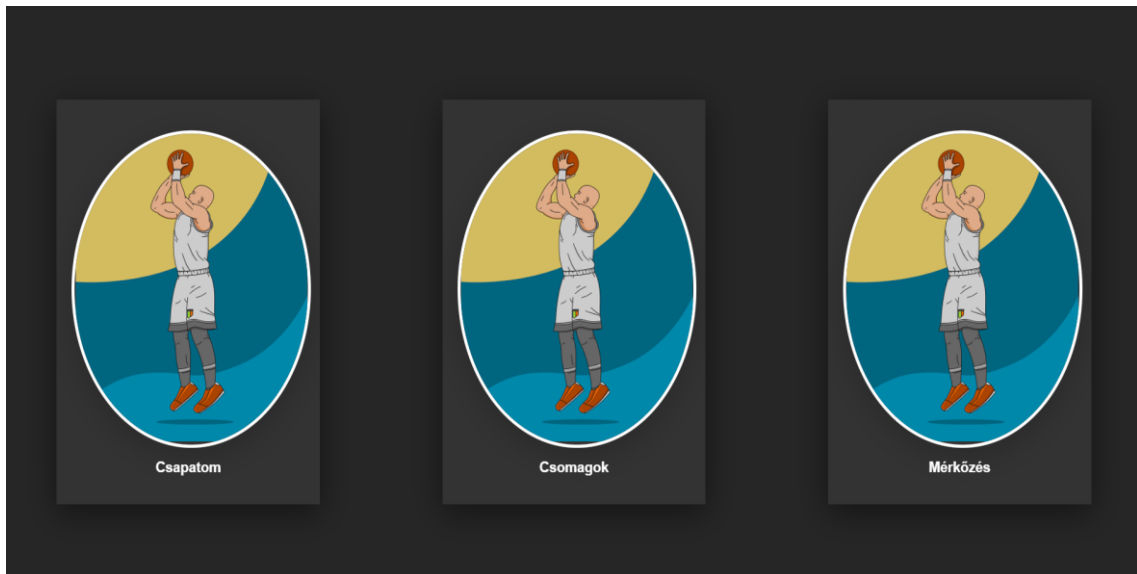
Az alkalmazásban elérhető funkciók nagymértékben függenek attól, hogy a bejelentkezett felhasználó milyen jogosultsággal rendelkezik. A későbbiekben ismertetem a funkciókat részletesen is, most csak a főbb elemeket említem meg.

Regisztráció után az alapértelmezett jogosultság a felhasználó. Egy másik lehetőség a moderátor. Aki ezzel rendelkezik számára elérhető az a lehetőség, hogy tudja változtatni a játékosok adatait. Emellett egy admin opció lesz még programomban, akinek jogában áll változtatni a felhasználók adatain.

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 megjeleníti az elérhetőséget, amire kell jelezni a problémát.

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

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.



1. ábra: Főmenü felületterv

Ezen a felületen emellett még meg fog jelenni a felhasználó csapatának neve és zsetonjainak száma is. Egy kijelentkezés gombot is elhelyezek a felületen.

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 tekintesse át őket.

A csomagok menüpont hasonlóan fog kinézni, mint a menü rész. Több lehetőség közül választhatja ki, mekkora értékűt szeretne venni. Ezután ki is nyitja.

A mérkőzés menüpont mérkőzések lejátszására szolgál. Itt meg lesz jelenítve a két csapat neve és az aktuális eredmény. Ezalatt látható lesz a játékosok kártyáinak listája.



2. ábra: Mérkőzés felületterv

A felhasználó a játékosra kattintással megmondja, hogy melyik játékosát szeretné, hogy párbajt vívjon az ellenfél egyik játékosával. A párbaj úgy zajlik, hogy mindkét játékos 3 pontszámát összegezzük, majd összehasonlítjuk. A győztes játékos szerez a csapatának egy pontot. Ezután újabb párbaj következik. A mérkőzések addig tartanak, míg 5 párbaj meg nem történt. A mérkőzések után visszatér a menübe.

Amennyiben moderátor jogosultsággal történik a bejelentkezés a menüben más opció is megjelenik. A megjelenő menüpontok közé bekerül a játékosok szerkesztése is. A szerkesztés mellett új elemek hozzáadására is lehetősége lesz a moderátornak. Lényegében a moderátori fiók a felhasználókkal kapcsolatos módosításokat nem tudja megtenni, viszont a játékban szereplő elemekkel mindenféle változtatást meg tud tenni.

Az admin jogosultsággal történő bejelentkezés után új menüpont fog megjelenni, viszont itt az opciók a felhasználókkal kapcsolatosak lesznek. A felhasználók szerkesztése menüpontnál lehetősége lesz módosítani a felhasználók nevét, a jogosultságukat, a zsetonjaik számát és lehetőségük lesz aktiválni, vagy inaktiválni az adott felhasználót. Mindezek mellett a felhasználóhoz tartozó csapat nevét is lehetősége lesz módosítani. Táblázatos formában fog megjelenni ez a felület.

Amennyiben a csomagok nyitásánál a kosárlabdázó már szerepel a felhasználó csapatában, az árának megfelelő zsetont kapja meg.

Regisztráció után a felhasználó kap öt darab kártyát és tízezer zsetont a vagyonához. Azért kap öt játékost, mert a mérkőzések során 5 játékosra lesz szüksége. A kezdeti vagyon azért lényeges, hogy már induláskor tudjon csomagokat nyitni a felhasználó.

Játékosok listája

név	összpontszám	hárompontos	zsákolás	ár	szerkesztés
név	összpontszám	hárompontos	zsákolás	ár	szerkesztés
név	összpontszám	hárompontos	zsákolás	ár	szerkesztés
név	összpontszám	hárompontos	zsákolás	ár	szerkesztés
név	összpontszám	hárompontos	zsákolás	ár	szerkesztés

3. ábra: Játékosok szerkesztése felületterv

A csomagok között többféle opció is megjelenik. Elérhető lesz 5000-150000 értékű csomag. Ezen csomagok kinyitásával a felhasználók bővíthetik csapataik létszámát.

Ezek a csomagok vagy játékost fognak tartalmazni, vagy zsetonokat, 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 a zsetonjaik számát nem tudják feltölteni, csak mérkőzések lejátszása után kapnak a mérkőzés kimenetelének megfelelő összeget, vagy csomagok nyitása útján szerezhetnek.

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 kezelése és mérkőzések lejátszása.

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

A *jatekosok* táblában találhatóak a játékban szereplő kosárlabdázók adatai. Itt tároljuk a kosárlabdázók nevét, 3 féle pontszámát (*osszPontszam*, *3pontos* és *zsakolas* mezők), árát és a hozzá tartozó képet. A képek tárolását a *kepek\jatekosok* mappában való-sítottam meg.

A *jogosultsagok* tábla a lehetséges jogosultságokat tartalmazza, ezek a következők: *admin*, *moderator*, *felhasznalo*.

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 táblában lévő *felhasznalok.id* mező idegen kulcs a *felhasznalok* tábla id mezőjére mutatva. A *jogosultsagok.id* mező szintén idegen kulcs, mely a *jogosultsagok* táblában lévő id mezőre mutat.

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 csere a kosárlabdázó. A táblában lévő *csapatok.id* mező idegen kulcsként funkcionál, a csapatok táblában lévő id mezőre mutat. A *jatekosok.id* mező szintén egy idegen kulcs, mely a *jatekosok* táblában lévő id mezőre mutat.

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 programom lényeges osztályait, függvényeit, metódusait. Ezáltal a logikai menete könnyebben érthető lesz más fejlesztők számára is.

Minden oldalt 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. A classes mappában található php fájlokban az oldalakhoz kapcsolódó adatbázis műveletek lettek elválasztva. 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 *bejelentkezesSession* 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 viszszaléptet 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 később 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 veszi 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, amennyiben a felhasználónak van elég zsetonja. A legolcsóbb esetén 1, a köztes esetén 2, a legdrágább esetén 3.

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 véletlenszerűen választ 0 és 1 között, majd a *tipus* változó veszi fel a kapott értéket. 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 90 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 játékost a kellő általános pontszámúak közül. Ellenben a *csomagAzon* session 0 értéket kap, vagyis pénznyeremény illeti a felhasználót. 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 azonosítójának értékével.

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 nézetben. Mindegyik kártya alján található egy párbaj gomb. A gomb lenyomására életbe lép a *btnParbaj* nevű action, ahol megtörténik a két párbajozó játékos kiválasztása a *sajatCsapat* és az *ellenfelCsapat* session értékeiből, majd a *parbaj* függvény meghívása a két játékos összesített pontszámát kapva paraméterül. 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 érték kerül a *sajatCsapat* és az *ellenfelCsapat* tömb esetén is. Ezen érték esetén a listázáskor az adott helyen szereplő elem nem jelenik meg. 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* vagy *dontetlen* függvények valamelyike rögzíti 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 *jatekoskartya* 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 elnevezésű függvények a jelenlegi adatokat adják vissza. A Módosítás gomb lenyomására lezajlik 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 *setJelszo* függvény végrehajtja ezt, amennyiben a jelenlegi jelszó valóban helyes. Az inaktíválás gombra kattintva a fiók inaktíválásra 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 átírnunk egy adatot és a módosítás gombra nyomunk megtörténik a *jatekosModositas* függvény futtatása, mely segítségével az adatok módosítása történik. 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 el lesznek mentve a *felv* névvel kezdődő sessionökben. Ezek kezdőértéke egy üres string.

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

Az *adatabazisclasses* 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 szolgált, mely segítségével a továbbfejlesztés gördülékenyebben tud haladni.

3.5. Tesztelési dokumentáció

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

A tesztelés során igyekeztem alkalmazni a fekete doboz és fehér doboz módszereket is, így tisztább képet kaptam a programom működéséről. Kezdetben 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 és nem ismerték a programom működését.

A legtöbb teszt a csomagok nyitására és a mérkőzésre vonatkozóan történt. Ezen tesztek során derült fény arra a hibára, hogy amikor 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 végeredmény 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.

The screenshot displays a web application for managing basketball players. In the background, a table lists players with columns for Name, Total Points, Three-pointers, Free Throws, Salary, and Photo. A 'Módosítás' (Edit) button is visible below the table. Overlaid on this is a modal window titled 'Játékos hozzáadása' (Add Player). The modal contains the following fields and constraints:

- Név** (Name): Text input, placeholder 'név'.
- Összpontszám** (Total Points): Text input, range '1 - 99'.
- Hárompontos** (Three-pointers): Text input, range '1 - 99'.
- Zsákolás** (Free Throws): Text input, range '1 - 99'.
- Ár** (Salary): Text input, range '1 - 250 000'.
- Kép** (Photo): Text input, placeholder '*.png vagy *.jpg'.

Below the input fields, a message states: 'Minden adat kitöltése kötelező! A pontszámok 1 és 99 között kell legyenek!' (All data entry is mandatory! The point counts must be between 1 and 99!). A blue 'Felvétel' (Add) button is at the bottom right of the modal.

5. ábra: Kosárlabdázók felvétele az adatbázisba

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 nézetben is történő tesztelésnek. Én, mint korábban is említettem a fejlesztés során a Google Chrome asztali és mobil felületét használtam. A tesztelés is ezek segítségével zajlott.

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, jobbra 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.

Emellett egy új piactér nevű menüpontot is szeretnék a jövőben elhelyezni az alkalmazásban, hogy a felhasználók megvehessék kedvenc játékosait.

A jövőben szeretném ezeket a fejlesztéseket megvalósítani az alkalmazásomban, ezáltal még jobbra 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 átgondoltam 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. Napjainkban nagyon sok ember ingázik a munkahelyére, vagy az iskolába. Ezalatt remek időtöltést nyújt egy ilyen játék. 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 mértékben 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 pár perc alatt 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. Ezt használva kinyílik számukra egy új világ, 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.

A program 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 rendszerek egyaránt alkalmasak 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. Android operációs rendszert futtató eszköz esetén Android 4.4, vagy újabb rendszer kell.

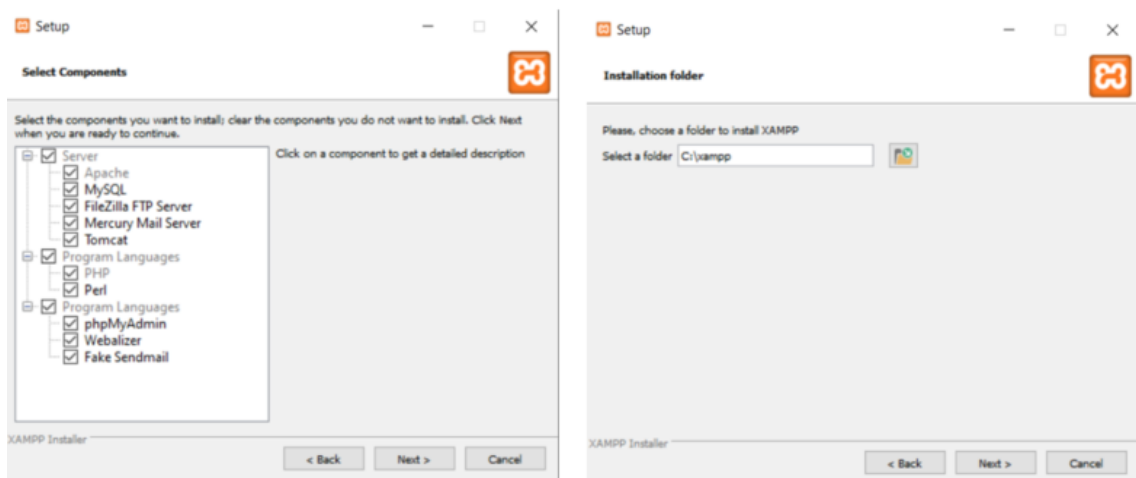
Amennyiben rendelkezik ilyen képességű eszközzel és telepítette a kellő szoftveket, akkor a program futtatása garantált.

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.

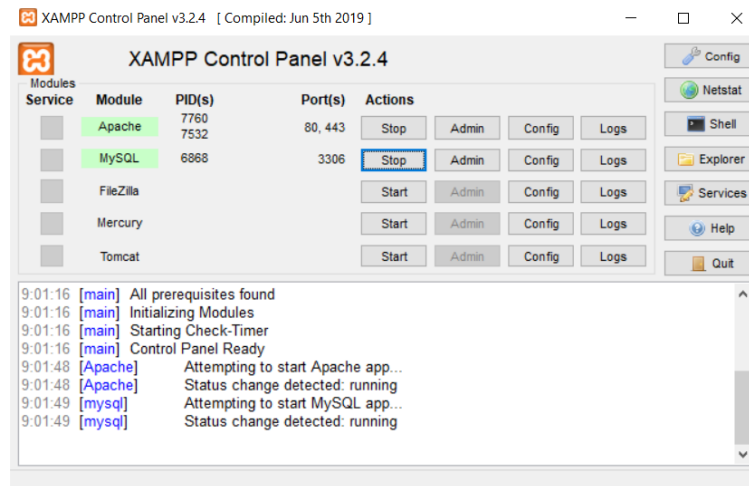
Első lépésként az XAMPP szoftvercsomag telepítése szükséges. Az XAMPP 7.4.3-as verzióját fogom használni. Miután elindítottuk a telepítő fájlt, engedélyt kér, hogy módosításokat hajthasson végre az eszközön. Az igen gombra kell kattintani. Ezután megjelenik az XAMPP telepítő üdvözlő része, a next gombra kell kattintani. Ekkor megjelenik egy felület, melyen kiválaszthatjuk, hogy a szoftvercsomag mely elemeit szeretnénk telepíteni. (6. ábra) 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. (6. ábra) Ezután a next gombbal lépünk 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.



6. ábra: XAMPP telepítés - modulok (balra) és telepítési hely választás (jobbra)

Amennyiben a vezérlőpanel nem indul el meg kell nyitnia a *xampp* mappában lévő *xampp-control* alkalmazást.

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



7. ábra: XAMPP vezérlőpanel, Apache és MySQL modulok elindított állapotban

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.

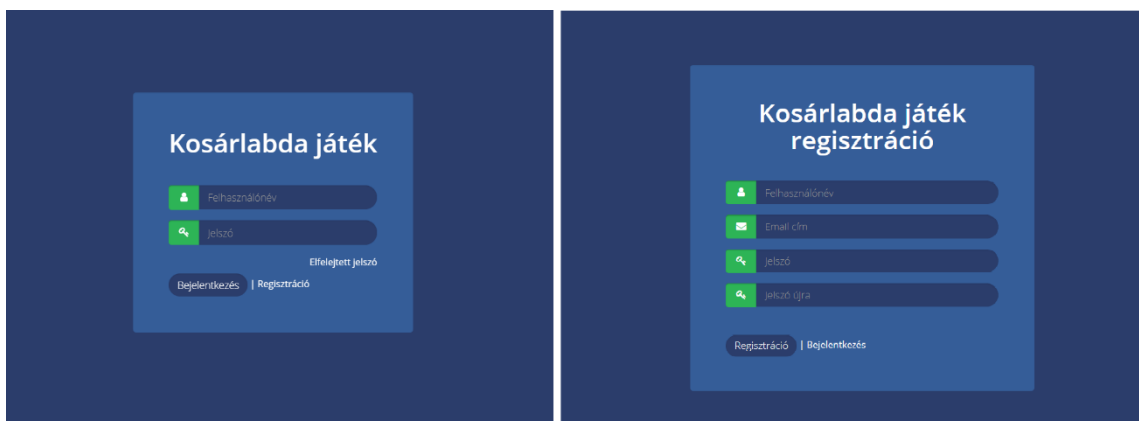
A *kosarlabdaapp* nevű mappát el kell helyeznünk a következő helyen: „*C:\xampp\htdocs*”. Amennyiben módosítottuk telepítéskor a helyet, úgy annak megfelelően kell elhelyeznünk a *htdocs* mappában.

Ezután a program indításához a böngésző címsorába kell írunk a következő szöveget: „*localhost/kosarlabdaapp/index.php*”. Ekkor elindul a program.

4.4. A program használatának a részletes leírása

Ebben a részben szeretném ismertetni az általam elkészített alkalmazás használatát. Az itt leírt lépéseket követve célnak megfelelően lehet használni a programot. A fejezetben bemutatom a programom minden funkcióját elejétől a végéig.

A program megnyitásakor a bejelentkezési felület fogad. Amennyiben rendelkezik már fiókkal, be tud jelentkezni a felhasználóneve és jelszava megadásával. Ha még nincs fiókja, kattintson a bejelentkezés gomb mellett elhelyezkedő regisztráció felíratra. Ezután egy másik felület fog megjelenni, mely a regisztrációt teszi lehetővé. Ehhez meg kell adnia egy felhasználónevet, egy email címet és egy jelszót, melyet újra be kell írnia. A jelszónak 10 karakternél hosszabbnak kell lennie. Ezután kattintson a regisztráció gombra. Sikeres regisztráció esetén a bejelentkezési oldal jelenik meg, emellett a felhasználó kap 5 játékost a csapatába és 15 ezer zsetont. A csapatneve a felhasználóneve lesz. Sikertelen regisztráció esetén egy hibaüzenet jelenik meg.



8. ábra: Bejelentkezési (balra) és regisztrációs (jobbra) felület

Bejelentkezés után a menü jelenik meg. A megjelenő menüpontok a felhasználó szerepétől függően láthatóak. Frissen regisztrált személy esetén *felhasznalo* a szerepkör. Ebben az esetben 4 menüpont lesz látható, melyek a következők: csapatom, csomagok, mérkőzés, fiókom beállítása. Ha a szerepköre *moderator*, akkor ezek mellett megjelenik egy játékosok szerkesztése menüpont is. Ha *admin* szerepkörrel rendelkezik, megjelenik egy felhasználók szerkesztése menüpont is. A felső sávban kiírásra kerül a bejelentkezett felhasználó zsetonjainak száma és csapatának neve.



9. ábra: Főmenü admin nézet

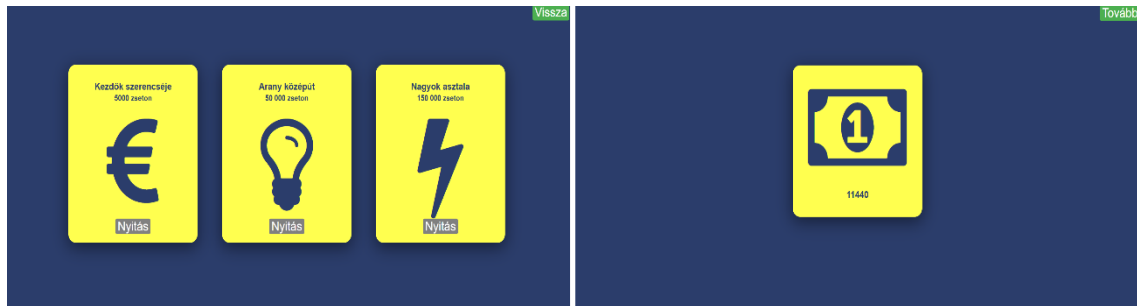
A jobb felső sarokban a legtöbb felületen található egy vissza, vagy egy tovább feliratú gomb. Ezek segítségével eljuthat az előző oldalra.

A csapatom menüpontot megnyitva megjelennek a felhasználó csapatában szereplő játékosok. két részben láthatóak: kezdőjátékosok és cserejátékosok. Lehetőség van a kezdőjátékosokat cserévé, a cserejátékosokat kezdővé tenni. A fő lényeg, hogy 5 kezdőjátékos legyen a csapatában, így fog tudni mérkőzéseket játszani. A mérkőzésen a kezdőjátékosok szerepelnek. Emellett a játékosok eladására is lehetősége van. Eladás esetén a játékos árának megfelelő összeget kapja a pénzéhez.



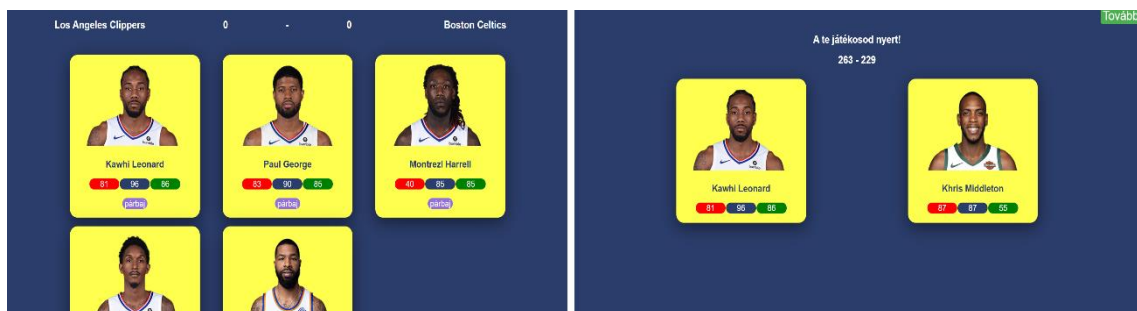
10. ábra: Saját csapat megjelenítése

A csomagok menüpontot megnyitva megjelennek az elérhető csomagok. Miután eldöntötte, hogy a 3 csomag közül melyiket szeretné kinyitni, kattintson a nyitás gombra. Egy csomag nyitása akkor lehetséges, ha van elég zsetonja. Sikeres nyitás esetén pénzt, vagy játékost kap. Ha a játékos már a csapatában van, akkor a program automatikusan eladja, vagyis az árának megfelelő mennyiségű zsetont kap. Ha a játékos még nem tagja a csapatának, akkor automatikusan hozzáadódik.



11. ábra: Csomagválasztó felület (balra) és egy kinyitott csomag tartalma (jobbra)

A mérkőzés menüpontra kattintva lehetősége lesz mérkőzést vívni egy másik véletlenszerűen választott csapat ellen. A mérkőzés a következőképpen zajlik: megjelennek a csapatában szereplő kezdőjátékosok, mindegyik kártyája alján egy párbaj gomb kap helyet. A felhasználó dönti el, hogy mikor melyik játékosal párbajozik. Az ellenfél csapatból véletlenszerűen történik az aktuálisan párbajozó kosárlabdázó kiválasztása a kezdőjátékosai közül. A mérkőzés addig tart, amíg nem használta fel mind az 5 kártyáját. A párbajok kimenetelét a kártyákon szereplő 3 pontszám összege határozza meg. Amelyik kosárlabdázó 3 pontszámának összege nagyobb, az nyeri a párbajt. Végül az eredmény megjelenítése történik.



12. ábra: Mérkőzés kiinduló állapot (balra) és párbaj (jobbra)

A fiókom beállítása menüpontra kattintva a saját adatait látja kiírva a regisztrációs felülethez hasonló módon. Lehetősége van módosítani a felhasználónevét, az email címét és a csapata nevét. A változtatás mentéséhez az adatok átírása után kattintson a módosítás gombra. A jelszó módosítása gombra kattintva megjelenik két új mező a felület alján, a

felsőbe a jelenlegi jelszavát írja, az alsóba az új jelszót, amire módosítani kívánja, majd a jóváhagyás gombra kattintva mentheti a beállítást. Az inaktíválás gombra kattintva a fiókját inaktívvá állítja, ezután nem fog tudni bejelentkezni.

Moderátor fiók esetén a játékosok szerkesztése menüpontra kattintva táblázatos formában megjelennek az adatbázisban szereplő kosárlabdázók az adataikkal együtt. Az adatok átírása után a módosítás gombra kattintva véglegesítheti a változtatást. A bal felső sarokban látható felvétel gomb segítségével új játékost adhat hozzá az adatbázishoz.

Játékos hozzáadása

Név:

Összpontszám:

Hárompontos:

Zsákolás:

Ár:

Kép:

Minden adat kitöltése kötelező!
A pontszámok 1 és 99 között kell legyenek!

Felvétel

Név	Összpontszám	Hárompontos	Zsákolás	Ár	Kép
Anthony Davis	98			42000	anthonydavis.png
DeMarcus Cousins	81			38000	demarcuscousins.png
JaVale McGee	80	43	90	37000	javalemcgee.png
Dwight Howard	80	54	85	32000	dwiththoward.png

13. ábra: Játékosok szerkesztése oldal megnyitott felvétel opcióval

Admin fiók esetén egy további menüpont is elérhető, mely a felhasználók szerkesztése. Ez a játékosok szerkesztésének felületével hasonló megjelenésű oldal. Az adatok átírását követően a sor végén található módosítás gombra kattintva mentheti az adatokat. Az inaktíválás gombra egy fiók inaktívvá tétele, míg az aktiválás gombra kattintva egy fiók aktívvá tétele zajlik. A fiókok szerepkörét legördülő menü segítségével lehetséges módosítani, de a módosítás gomb lenyomása mindenképp szükséges a véglegesítéshez.

A fent ismertetett információk fényében a játék rendeltetésszerű használata teljes mértékben megvalósítható, mivel minden oldal megjelenésére és az oldalon található funkciók működésére rávilágított ez a fejezet.

5. Összegzés

Ebben a fejezetben szeretném összefoglalni a munkámat, a szakmai hatását és a fejlesztés során szerzett tapasztalataimat. Mivel ez volt az első komolyabb általam fejlesztett alkalmazás, így rengeteget tanultam a fejlesztés során, amit remélem a jövőben fogok tudni hasznosítani.

Mivel ez egy komplex, összetett projekt, így a fájlstruktúra megvalósításánál igyekeztem nagy gondot fordítani arra, hogy a fejlesztés alatt és után is könnyen átlátható legyen a munkám, jól érthetőek legyenek a benne megvalósított folyamatok. Úgy érzem ezt sikerült megvalósítanom, azáltal, hogy mappákba szerveztem a projekt bizonyos részeit, vagyis az oldalak alapszerkezetét, a megjelenését és az adatbázisban történő műveleteket is külön helyeztem el.

Az adatbázis tervezés során több területet is jobban át kellett gondolnom a programom működésével kapcsolatban. Ez rendkívül hasznos volt, mivel így sok felesleges munkától óvtam meg magam. Az adatbázis struktúrájának megtervezése, majd elkészítése terén is komoly ismereteket szereztem.

A feladatom elkészítése során tapasztalatot tudtam szerezni a verziókezelő szoftverek használatában is, melyet szintén fontosnak tartok. A Git és GitHub párosát használtam a munkám eltárolására, mely nagy segítség volt számomra. Előfordult, hogy szükségessé vált korábbi verzióra visszatérni. Ekkor ezek a szoftverek megkönnyítették a helyzetemet.

A projekt során előfordult, hogy bizonyos műveletet később hajtottam végre, mint ideális lett volna a program helyes működéséhez, amire a tesztelés során fény derült, így tudtam javítani és fejleszteni magamat ezáltal is. A projekt későbbi szakaszában ez nagy hasznomra volt és a jövőre nézve is mindenképp hasznosnak láttam ezt a területet is, mivel így a fejlesztői gondolkodásom, látásmódom fejlődött.

A PHP programozási nyelvet is jobban megismertem, mivel a projekt nagy részében ezzel dolgoztam. Megismerkedhettem több egyedi függvénnyel ebben a nyelvben.

Az egyszerű, mégis stílusos megjelenést a CSS és a Bootstrap segítségével alakítottam ki. Ezeken a területeken is sikerült elmélyíteni az ismereteimet. Jobban átlátom, mit hogyan kell beállítanom a helyes megjelenítéshez.

Összességében úgy érzem a munkám több területen is a fejlődésemre volt. A jövőben minden biztonnal hasznát fogom venni az alkalmazás elkészítésére fordított munkaórák adta új ismereteimnek.

Forrásjegyzék

Bejelentkezési felület design: <https://github.com/truekasun/Simple-Responsive-Login-Page> (2019.12.12.)

Kártya design: <https://www.youtube.com/watch?v=pGEcBBjvRIU> (2020.01.10.)

Font Awesome 4: <https://fontawesome.com/v4.7.0/get-started/> (2020.03.15.)

Bootstrap: <https://getbootstrap.com/docs/4.0/getting-started/download/> (2020.03.15.)

jQuery.min.js: <https://code.jquery.com/jquery.min.js> (2020.03.15.)

Kosárlabdázók képei és adatai: <https://www.2kratings.com/current-teams-on-nba-2k20>,
<https://www.nba.com/players> (2020.02.15.)

Ábrajegyzék

1. ábra: Főmenü felületterv	10
2. ábra: Mérkőzés felületterv	10
3. ábra: Játékosok szerkesztése felületterv	11
4. ábra: Az adatbázis szerkezete	13
5. ábra: Kosárlabdázók felvétele az adatbázisba	20
6. ábra: XAMPP telepítés - modulok (balra) és telepítési hely választás (jobbra)	25
7. ábra: XAMPP vezérlőpanel, Apache és MySQL modulok elindított állapotban	26
8. ábra: Bejelentkezési (balra) és regisztrációs (jobbra) felület	27
9. ábra: Főmenü admin nézet	28
10. ábra: Saját csapat megjelenítése	28
11. ábra: Csomagválasztó felület (balra) és egy kinyitott csomag tartalma (jobbra)	29
12. ábra: Mérkőzés kiinduló állapot (balra) és párbaj (jobbra)	29
13. ábra: Játékosok szerkesztése oldal megnyitott felvétel opcióval	30