Debreceni SZC Beregszászi Pál

Technikum és Kollégium

Magyar Labdarúgás világa

Készítette:

Szabó Marcell

Tasnádi Richárd

Debrecen

2024

Tartalom

[Bevezető 3](#_Toc164860448)

[**1.1. Kik vagyunk:** 3](#_Toc164860449)

[**1.2. Projektfeladat témája: A Magyar labdarúgás** 3](#_Toc164860450)

[**1.3. Témaválasztás miértje, és céljai** 3](#_Toc164860451)

[Részletes feladat meghatározás 4](#_Toc164860452)

[**2.1. Szükséges hardverek, és szoftverek:** 4](#_Toc164860453)

[**2.2. Lehetőségek az oldalon regisztréció előtt és tudán:** 4](#_Toc164860454)

[**2.3. Regisztráció:** 4](#_Toc164860455)

[**2.4. Regisztrációhoz szükséges adatok:** 4](#_Toc164860456)

[**2.5. A főoldal:** 6](#_Toc164860457)

[**2.6. A játékosokról és csapataikról:** 6](#_Toc164860458)

[**2.7. Memória játék:** 7](#_Toc164860459)

[**2.8. Napi kviz:** 7](#_Toc164860460)

[**2.9. Tabella és góllövőlista:** 8](#_Toc164860461)

[**2.10. Az eredmények:** 9](#_Toc164860462)

[**2.11. Képek a csapatok stadionjairól:** 9](#_Toc164860463)

[**2.12. Profil módosítása:** 10](#_Toc164860464)

[Kivitelezés I. (Backend-et Készítette: Szabó Marcell) 11](#_Toc164860465)

[**3.1. felhasznált szoftverek (MySql, PHP, VSCodem, JS, guthub, CSS):** 11](#_Toc164860466)

[**3.2. Milyen osztályokat hoztál létre, osztálydiagramm:** 12](#_Toc164860467)

[**3.3. globális változók($\_Session[]…):** 13](#_Toc164860468)

[**3.4. fontosabb eljárások:** 14](#_Toc164860469)

[**3.5. Adatszerkezet(ER diagramm):** 15](#_Toc164860470)

[Kivitelezés II. (Asztali/Frontend-et Készítette: Tasnádi Richárd) 16](#_Toc164860471)

[**4.1. Felhasznált szoftverek:** 16](#_Toc164860472)

[**4.2. Felhasznált programozási nyelvek:** 16](#_Toc164860473)

[**4.3. Főoldal:** 16](#_Toc164860474)

[**4.4. Magyar játékosokról és csapataikról:** 20](#_Toc164860475)

[**4.5. Videók és képek:** 23](#_Toc164860476)

[**4.6. Bajnokságról:** 25](#_Toc164860477)

[**4.7.Eredmények:** 30](#_Toc164860478)

[Tesztelés 31](#_Toc164860479)

[**5.1. Bejelentkezés:** 31](#_Toc164860480)

[**5.2. Regisztráció:** 31](#_Toc164860481)

[**5.3. Módosítás:** 32](#_Toc164860482)

[**5.4. Törlés:** 32](#_Toc164860483)

[**5.5. Vélemény:** 32](#_Toc164860484)

[**5.6. Vásárlás:** 33](#_Toc164860485)

[**5.7. Adattábla:** 33](#_Toc164860486)

[Értékelés, összefoglalás 34](#_Toc164860487)

[**6.1. Fejlesztési elérhetőségek:** 34](#_Toc164860488)

[**6.2. Problémák a fejlesztés folyamán:** 34](#_Toc164860489)

# Bevezető

## **1.1. Kik vagyunk:**

Debreceni SZC Beregszászi Technikum tanulójai vagyunk, második évesek. Szoftverfejlesztő és tesztelő szakon vagyunk. A projektünket Visual Studio Code-ban készítetük el. Egyéb segítség később lesznek említve.

## **1.2. Projektfeladat témája: A Magyar labdarúgás**

A záróvizsgánk beadandójához témának a labdarúgást választottuk azon belül is a magyar labdarúgást, így ebből adódóan a projektünk címének a “Magyar Labdarúgás Világa” címet adtunk, és ebben a dokumentációnkban ezt a weboldalt szeretnénk minél részletesebben bemutatni. Az oldal a magyar labdarúgás életébe ad részletes betekintést, több érdekes funkcióval ellátva.

## **1.3. Témaválasztás miértje, és céljai**

A világon talán az egyik legtöbb tartalom a sportokban van, nem véletlenül ülnek le az emberek milliói a tv elé amikor egy adott sportban éppen egy világbajnokság döntője van. Viszont a sportok között mindössze egy van, aminek a legfontosabb döntőin a tv elé nem milliók, hanem milliárdok ülnek le az pedig a bizony a labdarúgás. Ezért mikor a projektfeladat témáján gondolkoztunk arra jutottunk, hogy egy olyan weboldalt akarunk készíteni, aminek a témájának az egyik, hanem a legnagyobb elérése van a világon ezért kézenfekvő is volt a foci legyen a fő téma, hiszen tényleg megannyi téma és érdekesség van ebben a sportban, amik előtt akár több óra hosszát is ellehet időzni, ha kedve tartja az embernek. Persze az sem volt hátrány, hogy alapból mindketten érdeklődtünk a foci iránt eddig is, így igazából nem is volt semmiféle nézeteltérés közöttünk a témaválasztással kapcsolatban.

Viszont ha már fociról van szó, akkor úgy gondoltuk, hogy ne külföldi focival kapcsolatos oldal legyen, hanem inkább magyar focival, hiszen külföldön több száz igényes futballal foglalkozó weboldal is van, ahol tényleg szintén mindent megtalálhat az odalátogató személy, amit tudni akar a világ futballjával kapcsolatban. Ezért a célunk egy olyan oldal létrehozatala volt, ami valamivel rugalmasabb, mint a legtöbb magyar futballal foglalkozó weboldal, hiszen a legtöbb ilyen oldal nagyon bonyolultan van megoldva, ami nem teljesen átlátható mindenki számára. Ezért mindenképpen azt akartunk, hogy ez egy olyan oldal legyen, ami esetleg az informatikai dolgokhoz kevésbé értő emberek számára is használható legyen, és ne legyenek fennakadások az oldal használata közben.

Fő célok között volt az oldallal természetesen az is, hogy a megszokott komolyabb oldalakkal ellentétben néhol ez egy valamivel lazább, humorosabb hangulatú legyen, több érdekes funkcióval, játékkal ellátva. Ez a weboldal a magyar focikedvelő emberek számára tökéletes arra, hogy részleteket tudjon meg azzal kapcsolatban, ami most éppen zajlik a magyar elsőosztályú labdarúgó bajnokságban, vagy éppen amblokk a Magyarfutballban. Az illető, ha tudni akar bármi információt a kedvenc magyar focicsapatáról, vagy bármelyik másik csapatról, akkor itt az alapvető tudnivalókat megtalálja, például a bajnokságban lévő csapatok stadionjai, becenevei, alapítási éve, stb. Természetesen, ha az oldalra látogató személy egy kis szórakozásra vágyik, akkor találhat azért magának egy két szórakoztató játékot, amivel elütheti az időt. A weboldal fő funkciója egyértelműen a jegyvásárlás, ami a megszokott oldalakhoz képest sokkal gördülékenyebben megoldható.

# Részletes feladat meghatározás

## **2.1. Szükséges hardverek, és szoftverek:**

Az oldal eléréséhez szükséges egy internetkapcsolattal rendelkező számítógép, és egy böngésző, mint például a google chrome, mozilla firefox, opera, stb.

## **2.2. Lehetőségek az oldalon regisztréció előtt és tudán:**

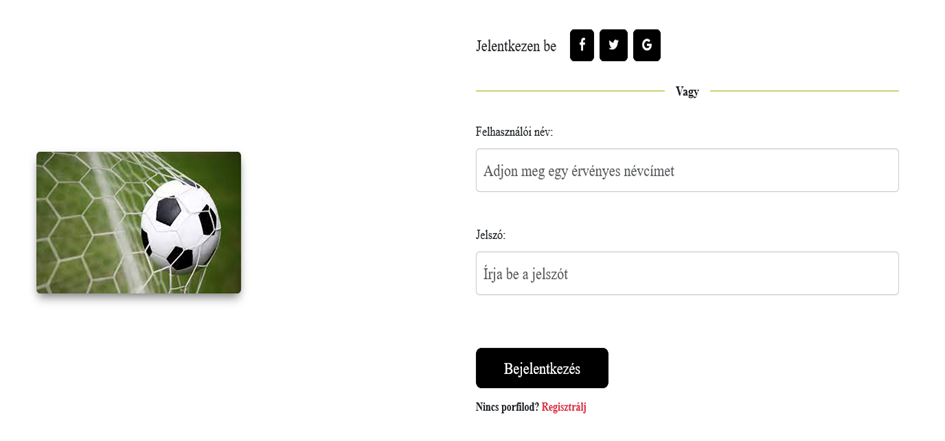
Regisztráció mindenképp kell, hiszen anélkül egyáltalán nem lehet belépni az oldalra, és elérni a szolgáltatásokat.

**Lehetőségek a regisztráció után:**

1. Jegyvásárlás a bajnoki mérkőzésekre
2. Focival, vagy akár egyéb sportokkal kapcsolatos hírek
3. Csapatokról, és játékosokról a legfontosabb információk
4. A Magyar bajnokság aktuális tabellája, és góllövőlistája
5. Az aktuális forduló eredményei
6. Memória játék, kvíz, videók
7. Profil módosítás

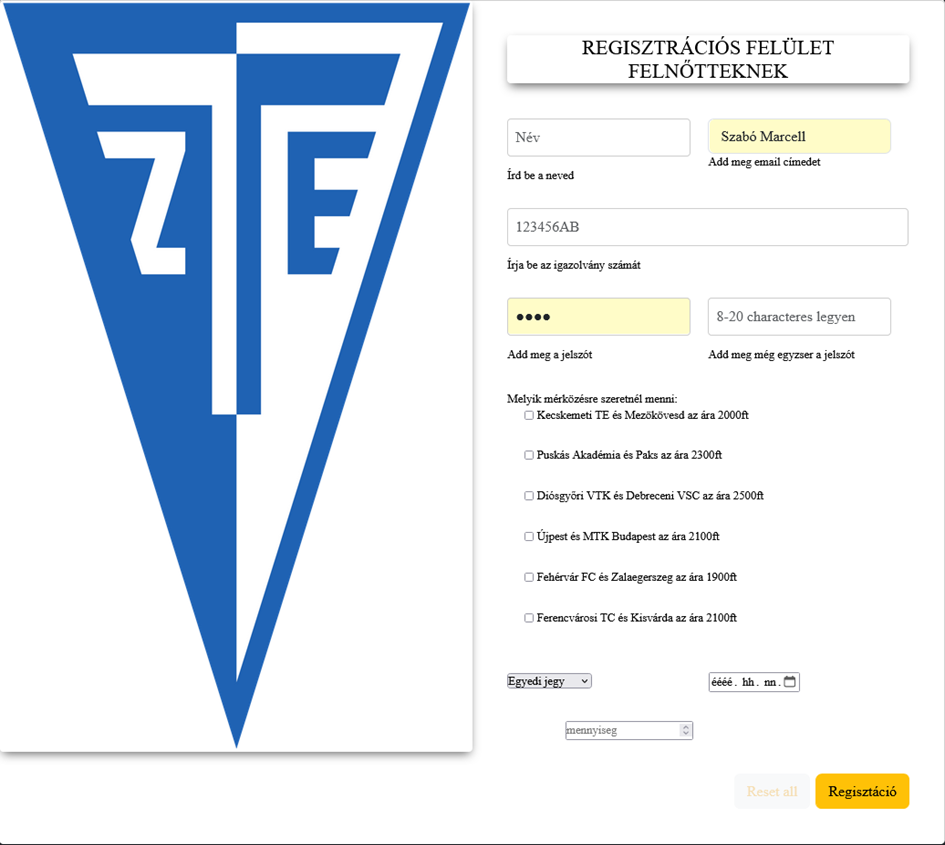
## **2.3. Regisztráció:**

Az oldal betöltésekor megjelenik egy bejelentkezési felület, ahol az oldalra látogató személy betud jelentkezni, ha van már fiókja. Viszont, ha nincs fiókja akkor a piros színnel jelzett regisztrálj szövegre kattintva tud létrehozni fiókot, amivel eltudja érni az oldal szolgáltatásait. Ha az adott személy úgy akar bejelentkezni, hogy már van fiókja, de rosszak az adatok akkor jelzi, hogy nem helyes a megadott jelszó, vagy felhasználó név, illetve felajánlja a regisztrációt, ha felhasználónak nincs érvényes fiókja az oldalra. A jegyvásárlás is a regisztrációs felületen érhető el.



## **2.4. Regisztrációhoz szükséges adatok:**

Az oldalt látogató embernek először megkell adnia a nevét, ami majd felhasználónév ként fog funkcionálni a bejelentkezésnél. Ezután szükséges lesz egy email cím. Majd ezek után kell az adott személy igazolványszáma a szokásos formában, ugye az első 6 karakternek számnak kell lennie, az utolsó kettőnek pedig valamilyen betűnek. Mindezek után megkell adni egy jelszót kétszer. Természetesen, ha a kettő nem egyezik azt egyből jelzi az embernek. Majd, ha ezek az alapvető adatok megvannak akkor kikell választania a személynek, hogy melyik adott mérkőzésre szeretne majd jegyet váltani. Itt megvannak adva a mérkőzések jegyárai is. Lehetősége van az illetőnek egyedi, vagy akár csoportos jegyet is váltani, emellett megtudja adni a jegyek számát, hogy mennyit szeretne vásárolni. Végül pedig az még megkell adni egy születési dátumot.



Ha megvan a regisztráció, és beengedte a személyt az oldalra akkor onnantól megnyílik a főoldal, és kedvére válogathat a lehetőségek közül az illető, hogy melyik szolgáltatását szeretni használni az oldalnak.

## **2.5. A főoldal:**

A főoldalon megtalálhatóak a legaktuálisabb futball hírek. Nagyrészt természetesen magyar futballal kapcsolatos hírek találhatóak a főoldalon, de azért vannak külföldi futballról is hírek. Sőt aki érdeklődik egyéb sportok iránt is, azok olvashatnak híreket kosárlabdával, röplabdával, vagy akár forma-1-el kapcsolatban is.

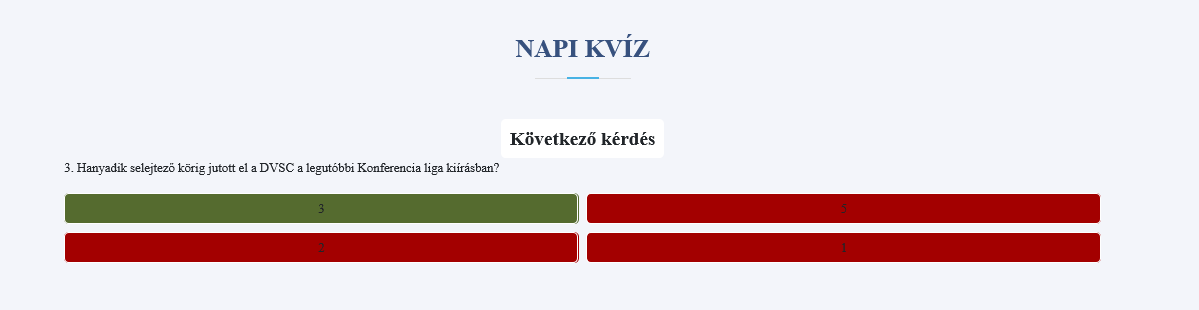
## **2.6. A játékosokról és csapataikról:**

****A weboldal ezen részén lehet olvasni a magyar futball legmeghatározóbb játékosainak történetéről, pályafutásáról. Azon oldal ezen részének másik fontos pontja ugye a csapatok, ahol csakis kizárólag a magyar elsőosztályú labdarúgó bajnokságban szereplői csapatokról találhat információt az ide látogató személy, aki a vagy a kedvenc csapatáról vagy bármelyik másik csapatról tudhatja meg a legfontosabb közérdekű információkat (mint a becenév, székhely, alapítási év, stb.)

## **2.7. Memória játék:**

Az oldalon lehetőségük van a szórakozni vágyó embereknek játszani egy kis memória játékot. Ebben a játékban kilenc darab magyar focistát kell párosítani egymással. Ha a két felfordított játékos nem egyezik egymással akkor visszafordulnak a kártyák, ha egyezik akkor pedig felfordítva maradnak a párosított kártyák, és alul pedig hozzáadja az elért pontokhoz. Aki újra szeretné játszani a memória játékot az a restart gombbal újrakezdheti.

## **2.8. Napi kviz:**

Aki az oldalra látogatók közül valami másfajta játékra vágyik, az a csapatinkról részlegen találhat egy 15 kérdésből álló napi kvízt a magyar futballal, de leginkább a magyar első osztályú labdarúgó bajnoksággal kapcsolatban. A kvíz kiírja az adott sorszámú kérdést, majd 4 lehetőség közül lehet választani. Minden kérdés megválaszolása után a “következő kérdés” gombra kattintva lehet megválaszolni a további kérdéseket. A kvíz a helyes választ zölddel jelzi, a helytelen válaszokat pedig pirossal. Aki újra szeretné próbálni a kvízt, az a kvíz végén az “újra kitöltöm” gombra kattintva elölről kezdheti a kérdés sort.

## **2.9. Tabella és góllövőlista:**

Akit érdekelnek a magyar bajnoksággal kapcsolatos további információk azok a csapatinkról részlegen megtalálhatják a bajnokság jelenlegi állását ábrázoló tabellát. Itt látható az adott helyen lévő csapatok logója. Mellettük a legfontosabb statisztikák, mint például a csapat eddig lejátszott meccseinek száma, a győzelmek, döntetlenek, és vereségek száma. Illetve, hogy hány pontot szerzett eddig az adott csapat-

Illetve megtalálhatóak a legtöbb gólt szerző játékosok is. Azok a játékosok akik azonos számú gólt szereztek azok ugyanazzal a helyezéssel vannak ellátva. Ezt a táblázatot is próbáltuk minél részletesebbre megcsinálni, ugyanis a játékosok neve mellett fel van tűntetve a játékosok csapata, nemzetisége, bajnokságban eddig lejátszott mérkőzése, természetesén eddig szerzett góljainak száma, és a gólpasszok is.

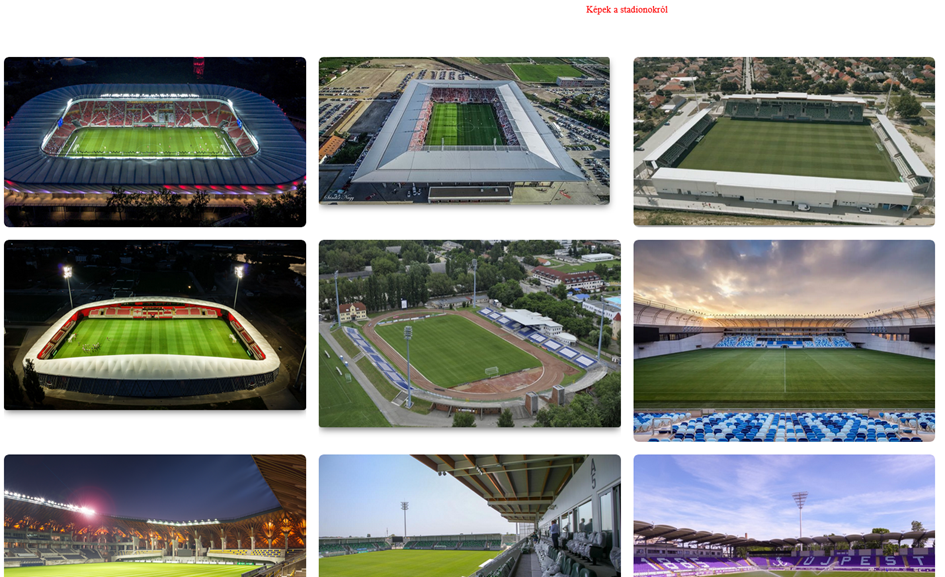
## **2.10. Az eredmények:**

Azok számára, akiknek nem elég ennyi statisztika az találhat még az oldalon egy “eredmények” nevű részleget, ahol a bajnokság adott fordulójának eredményei láthatóak. A képeken látható a bajnokság neve, és logója. Illetve természetesen felvan tüntetve a csapatok neve, és címere. Persze a mérkőzés időpontja is részletesen megvan adva, hogy melyik napon, és hány órakor kezdték a mérkőzést.

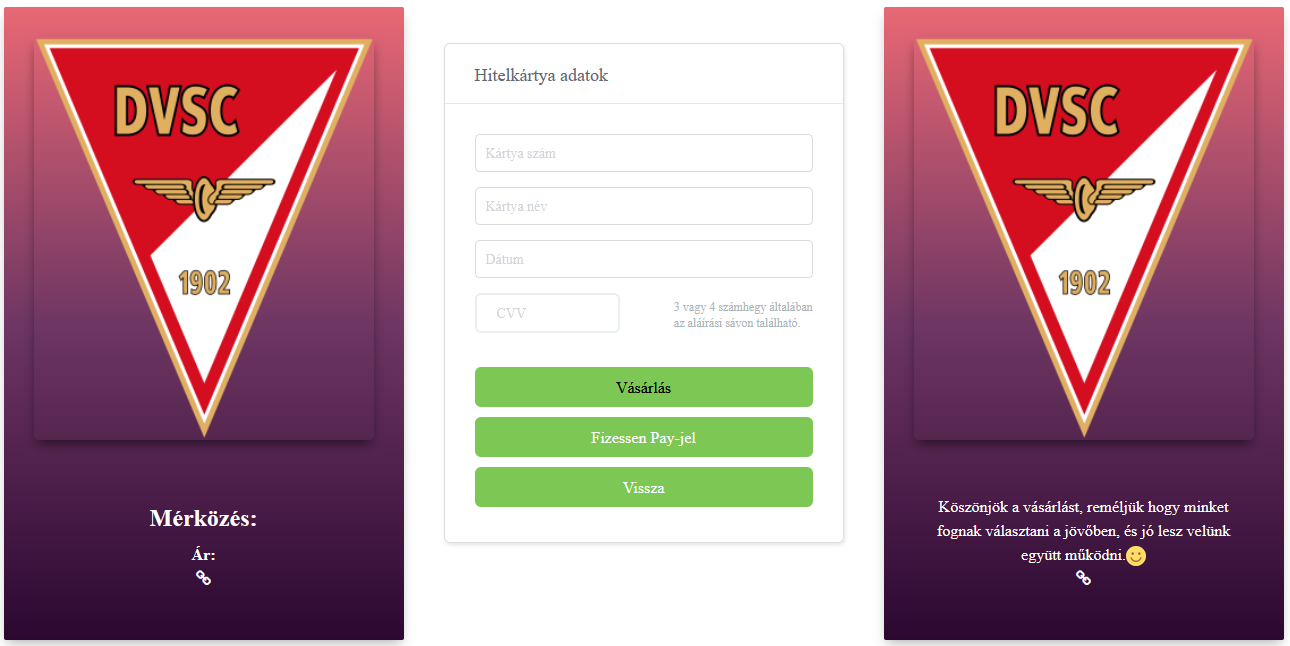


## **2.11. Képek a csapatok stadionjairól:**

A “Videók és képek” részen megtalálható a magyar első osztályú labdarúgó bajnokság 12 csapatának stadionja. Illetve a képek alatt megtalálható 2 kis videó, egyik egy kicsit humorosabb hangvételű. A második videó pedig 2-3 szép gólt tartalmazó, kicsit komolyabb videó.



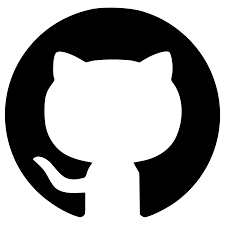
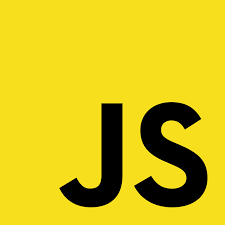
## **2.12. Profil módosítása:**

A profilom menüsávra kattintva a már meglévő profillal rendelkező személy módosíthatja profilján az adott dolgokat, ha úgy gondolja, hogy nem éppen helyesen vannak megadva az adatok. Itt lehet változtatni az igazolvány számát, az email címet. Illetve ha az illető egy másik mérkőzésre akar inkább jegyet váltani akkor tudja változtatni a mérkőzés időpontját, de akár a jegyek mennyiségét, és típusát is lehet változtatni. Ezek után a módosítás gombra kattintva ezek az adatok módosulnak. Ha az ember a már meglévő jegyét szeretné törölni, akkor megadja a már megvásárolt jegyek adatait, és a törlés gombbal törli a jegyeket.

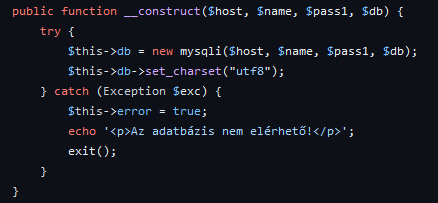
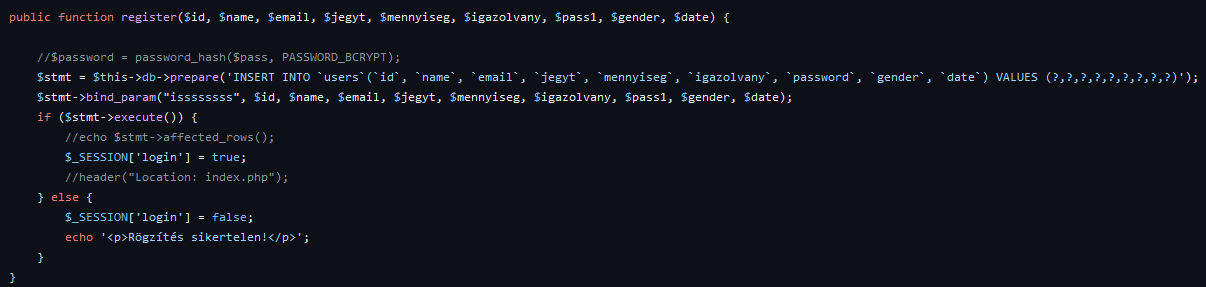
Természetesen újabb jegyek vásárlására is van lehetőség, itt csak megkell adni a szokásos adatokat (igazolványszám, email cím) majd az új meccs dátumát, jegyek mennyiségét, és típusát, ezután a vásárlás gombra kattintva egy új oldalra kerülünk ahol megkell adni a szokásos bankkártya adatokat (bankkártyaszám, kártyatulajdonos neve, kártya lejárati dátuma, és CVV kód. Majd a proceed gombra kattintva feldolgozásra kerül a vásárlás.

# Kivitelezés I. (Backend-et Készítette: Szabó Marcell)

## **3.1. felhasznált szoftverek:**

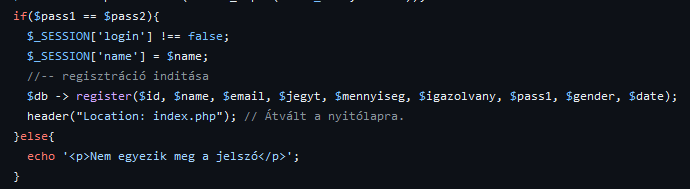
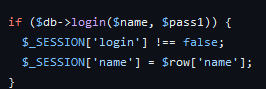
****A backendhez a MySql szerverét a PHP-t és a VSCode használatát vettem igénybe. Azért vettem ezeket igénybe, mert ezeket hallottam, hogy egyszerű a használatuk és népszerűek. A MySql arra a célra lett használva, hogy adatokat tudjunk felvinni, mint például a regisztrációhoz, a véleményadáshoz és a vásárláshoz. Nem volt egyszerű összerakni és még vannak gondolattok az elkészítésben, ami fejlesztés alatt állnak még. Azért használtam ezeket az alkalmazások, mert így tudtam létre hozni könnyen a regisztrációhoz a létrehozót kódokat és sok egyéb mást is. A MySQL adatbázis azért is volt jó, mert nem fejből kellett létre hozni a módosításhoz azt a kódot, ami az adatbázisban létre lett hozva a táblán belül. Így az osztályon belül létre lettek hozva a kódok és működő képes a módosítás, a törlés és a regisztráció. A php azért választottam, mert egyszerű volt összekötni az adott mappából a fájlokat és a hozzá tartozó kódokat. Volt olyan, hogy nem akarta a fájlokat elérni, próbáltam utána járni több nagyobb sikerrel, de nehezen összejött. A Visual Studio Code alkalmazást azért választottam, mert számomra egyszerű volt a használata és könnyebb is volt a githubra feltöltés a több mint száz soros kódokat. Azért is volt jó a github-ot használatátt igénybe venni, mert az előző verziót tudok frissíteni és nem mindig újra kezdeni („Verzió követés”). A js-t azért vettem igénybe, mert csináltam egy olyan videós váltós részt, amit 4 videó van és a 2 – 2 videó felváltva váltogatják egymást x idő után másik két videóra és majd újra a másik két videóra át kapcsolja. Nem biztos, hogy magában a php kódrészletben el tudtam volna érni ezt a lehetőséget, mint js-ben. Css-t segítségével tudtam egy olyan formázást létrehozni, a módosításhoz, a regisztrációhoz, a bejelentkezéshez, a vásárláshoz, a véleményhez és az egész oldalra, sok meghívót kódrészlettel mint például a td, p, az, img stb. Ezekhez egy olyan formázást, hoztuk létre, ami jó megjelenésben jeleníti meg a weboldalt egyes részeit. Sok minden kellett változtatni, de sikerült megoldani úgy, hogy megfelelően jelenjen meg a weboldalon a sok szöveg, kép, kvíz és ne legyen taszító a felhasználó részére.

## **3.2. Milyen osztályokat hoztál létre, osztálydiagramm:**

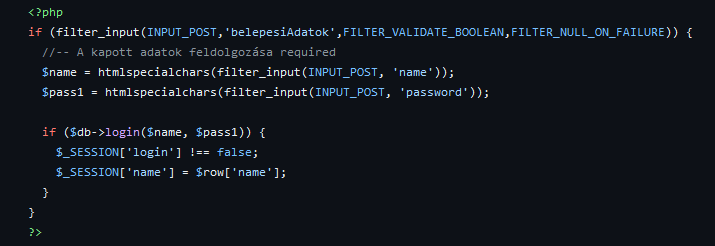
Egy osztály mappát hoztunk létre, ami benne van egy database.php fájl kód részletei. Itt function-nél hoztam létre egyéb lehetőségeket, mint például a login, a regisztráció, a módosítás, a törlés, a vásárlás és a véleményadás. Ezek a function segítségével tudtuk elérni azt a célt, hogy tudjuk működtetni a regisztrációt, a bejelentkezés módosítást és a törlést. Ez a database tárolás volt a legjobb lehetőség, hogy az adatok feltöltödjenek, és így meg tudtam oldani, hogy el tudja érni a PHP fájlba az egyéb mysql funkciókat, mint például a törlés, módosítást stb. Nem voltak egyszerűek létre hozni ezeket a funtion-et, kellett egy kicsit gondolkodni, mit is akarok elérni az osztállyal. A function constructor ez azt vizsgája, hogy elérhető e az adatbázis a mysql-ből. Ha nem érhető el akkor kiíratja, hogy az adatbázis nem elérhető. Ha mégis elérhető akkor el tudja érni a adatbázist és utf8 kódolással tudja olvastatni a regisztrált adatokat.

A regisztrációnál létre lett hozva a $stmt ami segítségével a prepare lekérdezésével tudtam azt a célt elérni a működését, hogy az adatbázissal mi alapján érje el a megadaott regisztrált adatokat, hogy az adatokat fel lehesn tölteni és minden zökkenő mentessen működjön. Létre lett hozva egy $stmt vizsgálat, ha a név létezik azzokkal az adatokkal, amikor egy felhasználó regisztrált, akkor sikertelen a regisztáció. Ha mégse létezik akkor a regisztráció sikeres és tovább küldi a weboldalra.

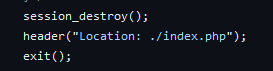
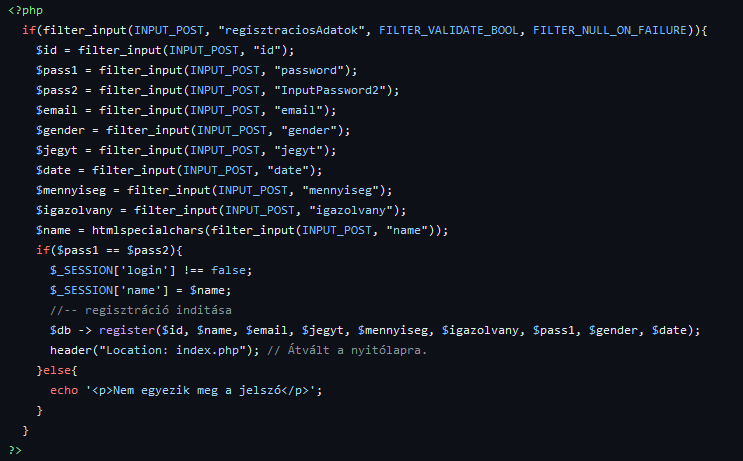
## **3.3. globális változók:**

A képen szöveg, képernyőkép, képernyő, szoftver látható

Automatikusan generált leírásGlobális változókat, ami a project során használtam a $\_SESSION, változók, echo használata, a GET, a POST, a DELETE és az UPDATE és logikai váltózokat. Ezek használatával és segítségével tudtam elérni azt a lehetőséget, hogy az oldal tudja a CRUD használatát és könnyebb legyen a kezelhetőség az adott személynek. A $\_SESSION -t leginkább a login és a regisztráció működtetését lett le kódolva. Az a cél volt ezzel a SESSION-nel, hogy a login meg a neve-t vizsgálja és az alapján engedje be majd a felhasználót. A képelátszódik is hogy mit vizsgál a $\_SESSION-ben. Adatbázisból kéri le az adott személyhez a neveket. A regisztrációnál is létre lett hozva ott is a $\_SESSION és megvizsgálja a neve alaján a dolgokat ha megegyezik a regisztráció megegyezik és létre lett hozva egy logikati vizsgálat is, hogy megfelelő a az adott proilnál a regisztráció. A bejentkezést pedig a név alapján vizsgálja, hogy létezik-e az adott felhasználóid név. Ha léteszik akkor nem engedi létre hozni az adott név igazolvány és email cím alapján, hanem más igazolvány számot, nevet és email címet kell kitaláljon a felhasználó.

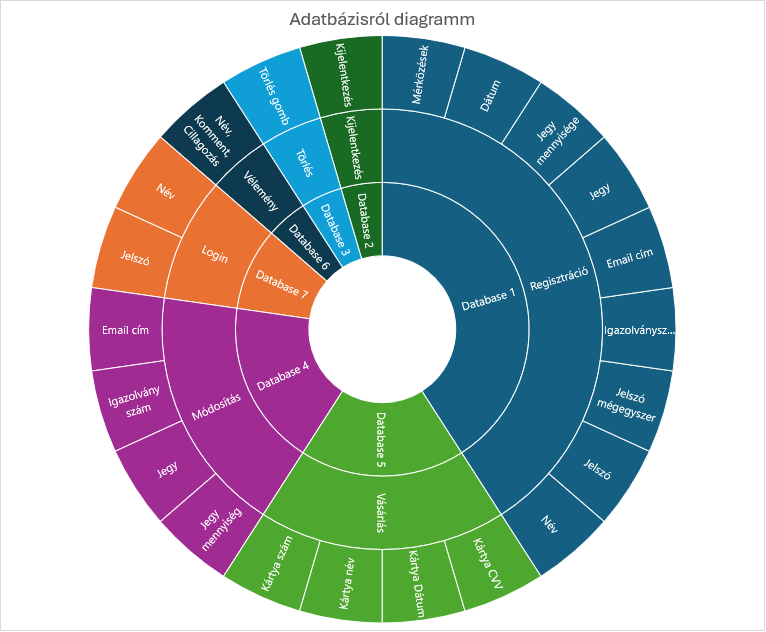


## **3.4. fontosabb eljárások:**

Fontos eljárások, amit igénybe vettünk, a login, a regisztráció, a módosítás és a törlés. Az első kép a loginra szolgáló kódrészlet, ahogy létre lett hozva, hogy hogyan olvastassa ki az adatbázisból a nevet és hogyan tudja ellenőrizni. A második kép regisztrációs fontosabb eljárásokat van egy kód rész, amiket el szerettünk érni a felhasználótól. Például az igazolványszám, jegy itt arra gondolunk, hogy csoportos vagy egyéni jegyet akar és a jegynek van egy mennyiség számlálója is, hogy mennyit szeretne kérni például

10-t, 100-t akár 1000-t szeretne. Nem volt ez se egyszerű létre hozni és megoldani, amikor a módosításhoz akartuk működtetni. A harmadik kép a módosításnál próbáltuk azokat létrehozni, amik fontosabb lehetnek a módosításra. Mint például az igazolványszám, a dátum, a jegy, a jegy mennyisége és az email cím. És így, ha módosítja vagy nem módosítja akkor meg tudja majd vásárolni. Később lesz bővebben kitérés. A kijelentkezés php kódrészletből annyiból áll a kód részlet látszik, amin a kis képen látható, hogy kilépteti az index.php alapján. Tökéletesen megy és a login felületre fogja a felhasználót léptetni automatikusan.

## **3.5. Adatszerkezet (diagramm):**

A felhasználói rendszer része lett létre hozva a regisztráció, be- és kijelentkezés, adatlap és adatlap módosítása. Kereskedelmi rendszer is létre lett hozva, ez a vásárlás lenne. Ezzel a vásárlással azt a cél akartuk elérni, hogy amikor módosította az adatlapját akkor megtudja vásárolni a jegyet, ha nem akarja módosítani, akkor a vásárlást abba az állapotban tudja megvásárolni, ahogy regisztrált és így is megvásárolhatja a megadott adatokat. A törlés pedig úgy lett megoldva, hogy ha rá kattint a törlés gombra, akkor a felhasználói profilja törlődik és a login felületre fogja dobni, ahonnan újra kell kezdje az egész profil létrehozását, ha rá kattint a regisztráció létrehozására. A jövőben még szóba jöhet, hogy online felületre is elkészítjük a felületet csak még meg kell oldjuk rajta pár dolgokat, hogy normális kinézet legyen és minden tökéletes legyen a felhasználói élmény.

# Kivitelezés II. (Asztali/Frontend-et Készítette: Tasnádi Richárd)

## **4.1. Felhasznált szoftverek:**

1. **Windows 10:** A szakdolgozat elkészítéséhez a Windows 10 szoftvert használtam, hiszen a világon jelenleg mindenki szinte ezt használja, és ez az egyik legfejlettebb **is.**
2. **Visual Studio Code:** A kódszerkesztéshez szintén a szakmában a leghasználtabb programot választottam, a Visual Studio Codeot. Legfőképp azért, mert rengeteg billentyűzet kombináció és segítség van benne, ami könnyebbé teszi a munkánkat.
3. **XAMPP:** Az oldal elkészítéséhez természetesen elengedhetetlen volt, hogy egy olyan olyan programot használjak, amivel tudom ellenőrizni az addig készített munkámat.

## **4.2. Felhasznált programozási nyelvek:**

1. **HTML: HyperText Markup Language:** A weboldalak elkészítéshez leggyakrabban alkalmazott programozási nyelv, hiszen nagyon könnyen használható.
2. **PHP: HyperText Processor:** A php egy nyílt forráskódú, széles körben használt, általános célú, és szerver oldali programozási nyelv. Különösen jó webfejlesztési támogatással és HTML-be ágyazási képességekkel.
3. **CSS:** Ez a programozási nyelv a weblap vizualitásáért felel. Az oldal megjelenítésére használt böngésző megvizsgálja a css kódban elhelyezett tulajdonságokat, és az alapján jelenítik meg a weboldalt.
4. **JavaScript:** A JavaScript egy objektumorientált, kliensoldali script nyelv. Csak olyan dolgokat tudunk vele elintézni, ami a felhasználó oldalán jelenik meg.

## **4.3. Főoldal:**

1. <section id="posts" class="posts">: Ez egy szekció, amely a főbb híreket jeleníti meg az oldalon, ennek azonosítója "posts", osztályneve pedig "posts".
2. <div class="container" data-aos="fade-up">: Ez egy konténer, amely az "aos" adatkarakterisztikát használja, hogy animációt hozzon létre a "fade-up" értékkel.
3. A szekcióban két további konténer található, amelyekben a bejegyzések elhelyezkednek.
4. A bejegyzésekhez tartozó címek, képek és szövegek az <a>, <img>, és <h2> elemekkel vannak megjelenítve.
5. A bejegyzésekhez kapcsolódó metaadatok, mint a kategória, a dátum és az író a <div class="post-meta"> és <div class="author"> elemekkel vannak megjelenítve.
6. A bejegyzések közötti határokhoz <hr> elemeket használtam.

A kód ezen része arra szolgál, hogy formázott módon megjelenítse a legfontosabb híreket a magyar fociról, képeket és hozzájuk kapcsolódó információkat, valamint hogy biztosítsa a megfelelő elrendezést és stílust a weboldalon.

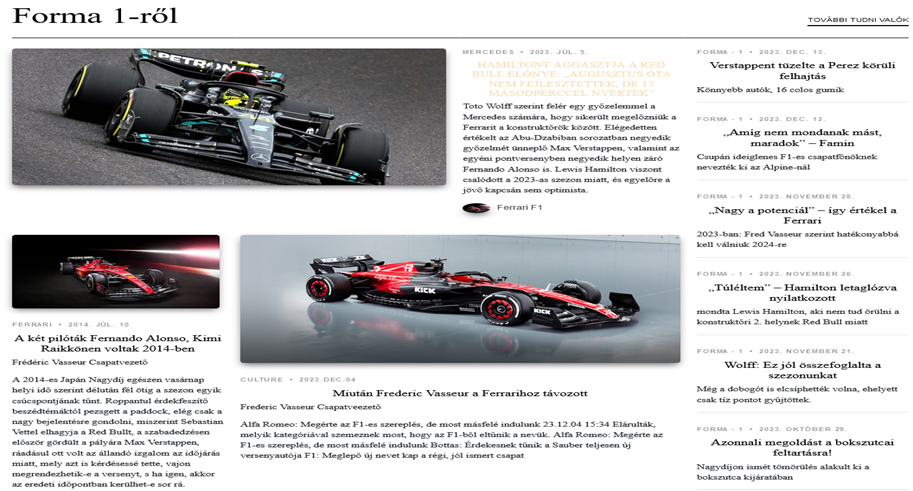
1. <div class="trending">: Ez a div elem határozza meg a "Trending" szekciót.
2. <h3>Trending</h3>: Ez a cím jelzi a felhasználóknak, hogy ez a rész a népszerű témákat vagy bejegyzéseket tartalmazza.
3. <ul class="trending-post">: Ez egy rendezett lista, amely tartalmazza a népszerű bejegyzéseket vagy témákat.
4. Minden bejegyzéshez egy listaelem (<li>) tartozik, amely tartalmazza:
   * <a href="#">: Ez egy hivatkozás a bejegyzésre vagy témára.
   * <span class="number"></span>: Ez a szám jelzi, hogy hányadik a sorban a bejegyzés.
   * <h3>: Ez a cím jelzi a bejegyzés vagy téma címét.
   * <span class="author">: Ez a szerző nevét jelzi a bejegyzés vagy téma mellett.
5. A href="#" attribútum azt jelzi, hogy a hivatkozásnak nincs valós URL-je, ezáltal nem vezet sehová.

Ez a kód egy "Trending" szekciót határoz meg, amely népszerű vagy aktuális témákat jelenít meg a weboldalon.

Ez a kód arra szolgál, hogy egy listában megjelenítse a népszerű témákat vagy bejegyzéseket a weboldalon, és lehetővé teszi a felhasználók számára, hogy ezeket megtekintsék vagy rákattintsanak rájuk.

Ez a HTML kód egy kategória szekciót határoz meg, amely a Forma 1 világával kapcsolatos tartalmakat jelenít meg. Néhány fontos elem:

1. <section class="category-section">: Ez egy szekció, amely a "category-section" osztályhoz tartozik.
2. <div class="container" data-aos="fade-up">: Ez egy konténer, amely az "aos" adatkarakterisztikát használja, hogy animációt hozzon létre a "fade-up" értékkel.
3. <div class="section-header d-flex justify-content-between align-items-center mb-4">: Ez a szekció fejléce, amely a szekció címét és egy "További tudni valók" linket tartalmaz.
4. A tartalom részben megjelenített bejegyzések, amelyeknek adatait, címeit, képeit és szövegeit a különböző HTML elemekkel adjuk meg, például <a>, <img>, <h2>, <p>, <div>.
5. A bejegyzések mellett megjelenített metaadatok, mint például a dátum és az író neve, a <div class="post-meta"> és <span class="author"> elemek segítségével jelennek meg.
6. A kategória szekcióban további bejegyzések vannak elhelyezve a bal oldalon, amelyeket egy col-md-9 oszlop tartalmaz, valamint a jobb oldalon, egy col-md-3 oszlopban, amelyek felsoroljuk az aktuális Forma 1 témákkal kapcsolatos híreket.

Ez a kód arra szolgál, hogy egy jól strukturált módon megjelenítse a Forma 1-hez kapcsolódó tartalmakat, és lehetővé teszi a felhasználók számára, hogy könnyen böngésszenek és olvassanak az aktuális témákról és eseményekről.

Ez a HTML kód egy másik kategória szekciót határoz meg, amely a kosárlabdával és a röplabdával kapcsolatos tartalmakat jeleníti meg. Néhány fontos elem:

1. <section class="category-section">: Ez egy szekció, amely a "category-section" osztályhoz tartozik.
2. <div class="container" data-aos="fade-up">: Ez egy konténer, amely az "aos" adatkarakterisztikát használja, hogy animációt hozzon létre a "fade-up" értékkel.
3. <div class="section-header d-flex justify-content-between align-items-center mb-4">: Ez a szekció fejléce, amely a szekció címét és egy "Bövebb információk" linket tartalmaz.
4. A tartalom részben megjelenített bejegyzések, amelyeknek adatait, címeit, képeit és szövegeit a különböző HTML elemekkel adja meg, például <a>, <img>, <h2>, <p>, <div>.
5. A bejegyzések mellett megjelenített metaadatok, mint például a dátum és az író neve, a <div class="post-meta"> és <span class="author"> elemek segítségével jelennek meg.
6. A kategória szekcióban további bejegyzések vannak elhelyezve a bal oldalon, amelyeket egy col-md-9 oszlop tartalmaz, valamint a jobb oldalon, egy col-md-3 oszlopban, amelyek felsorolják a legfrissebb kosárlabda és röplabda híreket.

Ez a kód arra szolgál, hogy egy jól strukturált módon megjelenítse a kosárlabda és röplabda témájú tartalmakat, és lehetővé teszi a felhasználók számára, hogy könnyen böngésszenek és olvassanak az aktuális témákról és eseményekről.

Ez a kód egy kapcsolatfelvételi űrlapot és elérhetőségi információkat tartalmazó szekciót határoz meg. Néhány fontos elem:

1. <section id="contact" class="contact text-center">: Ez egy szekció, amelynek azonosítója "contact", és osztályai között szerepel a "contact" és a "text-center".
2. <div class="container" data-aos="fade-up">: Ez egy konténer, amely az "aos" adatkarakterisztikát használja, hogy animációt hozzon létre a "fade-up" értékkel.
3. <div class="section-header">: Ez a szekció fejlécét tartalmazza, amely a címet és egy rövid leírást jeleníti meg a kapcsolatfelvételről.
4. Az elérhetőségi információkat tartalmazó <div class="row gy-4"> rész a konténer alatt van. Minden információ egy-egy "col-md-6" oszlopban van elhelyezve, így két információ van egy sorban.
5. Az információkat tartalmazó oszlopokat "info-item" osztállyal látjuk el, és mindegyiknek van egy ikonja is, amely az információtípushoz kapcsolódik. Az ikonokhoz a Font Awesome ikonkészletet használják, mint például a "fa fa-map", "fa fa-envelope", "fa fa-phone", "fa fa-share".
6. Az "Helyszín", "Email Cím", "Telefonszám" és "Nyitva tartás" információkat jelenítik meg, különböző színekkel és ikonokkal kiemelve.

Ez a kód egy jól strukturált kapcsolatfelvételi szekciót határoz meg, amely segítségével az ügyfelek könnyen megtalálhatják a velünk való kapcsolatfelvétel módját és az elérhetőségi információkat. Az interaktivitást Google térkép beágyazásával növeltük, amely lehetővé teszi az ügyfelek számára, hogy megnézzék a címünket, és környékét.

## **4.4. Magyar játékosokról és csapataikról:**

A HTML kód ezen része a játékosokról részleget mutatja be.

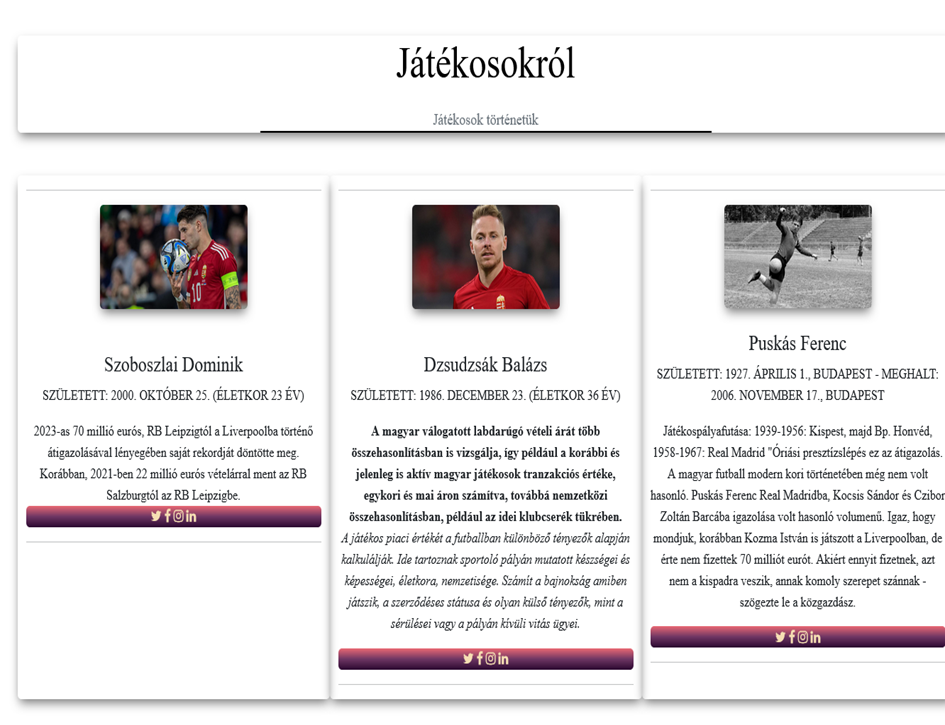
1. A kód egy `<section>` elemmel kezdődik, amely egy nagyobb tartalomrészletet reprezentál az oldalon.

2. A tartalom egy `<div>` elemmel kezdődik, amelynek osztályneve "container".

3. A képernyő középre igazított tartalmat tartalmazó további `<div>` elemek következnek. Ezeknek az osztályneve "col-12 text-center mb-5" és "col-lg-4 text-center mb-5".

4. Minden játékoshoz tartozik egy kép, név, születési dátum és egy rövid szöveges leírás. Ezek az adatok rendezett módon vannak elrendezve.

5. A játékosokhoz tartozó szociális média linkek ikonok formájában találhatók meg, amelyek egy-egy szociális platformot képviselnek (Twitter, Facebook, Instagram, LinkedIn).



Ez egy további HTML kód részlet, amely a weboldal egy másik részét tartalmazza. Ez a rész a "csapatok" részleteit mutatja be.

A kód egy <div> elemmel kezdődik, amelynek osztályneve "row gy-4 portfolio-container". Ez a Bootstrap alapú elrendezés egy olyan tartalomtartó rész, amely igazából egy portfóliót tartalmaz.

Minden egyes csapatot bemutató rész egy <div> elembe van csomagolva, amelynek osztályneve "col-lg-4 col-md-6 portfolio-item", ami azt jelzi, hogy ezek a részletek egy négyoszlopos elrendezésben jelennek meg nagyobb kijelzőkön és egy hatszoros oszlopos elrendezésben kisebb kijelzőkön.

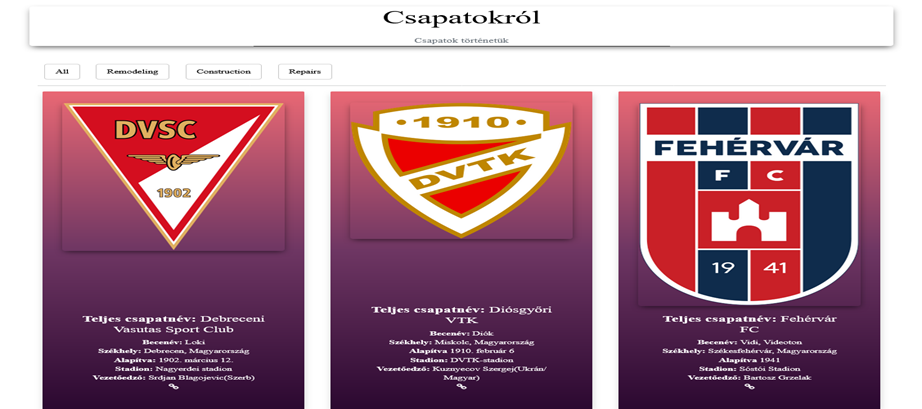
Minden csapatnak van egy képe, valamint információk a teljes csapatnévről, a becenévről, a székhelyről, az alapítási dátumról, a stadionról és a vezetőedzőről.

A képek útvonalai relatívak, és a "kepek/csapatok/" mappában vannak.

A <a> elemek segítségével a felhasználók további részleteket nézhetnek meg egy adott csapattal kapcsolatban, vagy megnézhetik a képeket nagyobb méretben.

A csapatokhoz különböző szűrők (pl. "filter-remodeling", "filter-construction") vannak hozzárendelve, amelyek kategorizálják őket adott jellemzők alapján, például a csapatok általános állapotuk vagy tevékenységük alapján.

Ez a kód egy dinamikus, vizuálisan vonzó módon mutatja be a magyar foci csapatokat, és lehetővé teszi a felhasználók számára, hogy részletes információkat kapjanak róluk, valamint képeket nézzenek meg.



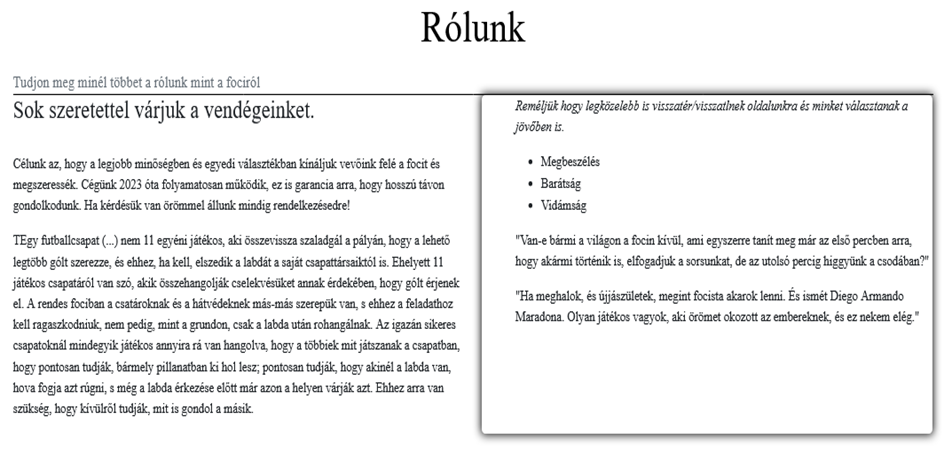
**About (Rólunk) szakasz:**

* + Az id="about" attribútummal rendelkező <section> elem a rólunk szakaszt határozza meg.
  + Ez a szakasz egy olyan blokkot tartalmaz, amely bemutatja az oldal készítőinek profilját
  + Tartalmaz egy címet ("Rólunk") és egy bevezető szöveget, valamint két olyan oszlopot, amelyekben további információk és képek találhatók.
  + Az első oszlop egy rövid üdvözlő szöveget, valamint egy képet és egy hosszabb leírást tartalmaz az oldal készítőiről.
  + A második oszlop tartalmaz egy további üzenetet, egy listát és további idézeteket, valamint egy képet, amelyet egy videó linkkel lehet megnyitni.

**Clients (Ügyfelek) szakasz:**

* + Az id="clients" attribútummal rendelkező <section> elem a "clients" szakaszt határozza meg.
  + Ez a szakasz egy olyan blokkot tartalmaz, amely az ügyfelek logóit vagy képeit jeleníti meg egy csúszkában.
  + A logók vagy képek egy Swiper slideren keresztül vannak megjelenítve, amely lehetővé teszi a felhasználók számára, hogy lapozzanak a képek között.
  + Minden ügyfél logója vagy képe egy-egy swiper-slide oszlopban van elhelyezve.

Ez a kód segít az oldal látogatóinak megismerni az oldal készítőit. A Swiper slider megjelenítése a logókkal vagy képekkel a szakasz vizuális vonzerejét növeli.

****

## **4.5. Videók és képek:**

Ez a kód egy galériát határoz meg, amely stadionokról készült képeket jelenít meg egy weboldalon.

**Cím és Szöveges Tartalom:**

- A `<h2>` címsor meghatározza a galéria címét: "Képek a stadionokról".

- A HTML kód tartalmaz egy kommentet (`<!-- ======= Gallery Section ======= -->`), ami segít a fejlesztőknek azonosítani a kódblokkot.

- A `<section>` elem `id="gallery"` attribútummal rendelkezik, ami lehetővé teszi a CSS vagy JavaScript kódban való hivatkozást.

- A galéria elemeket egy `container-fluid` osztályú `<div>` elem tartalmazza, ami teljes szélességű konténert biztosít a galéria számára.

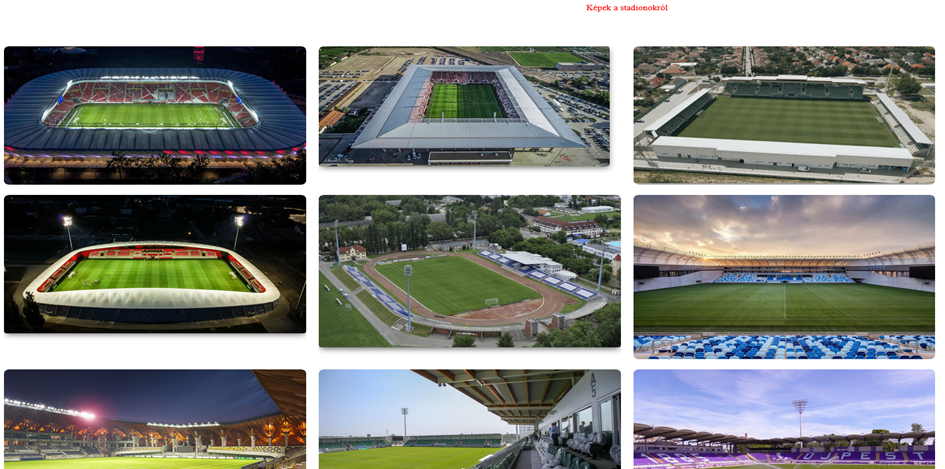
- A képek egy `<div class="row gy-4 justify-content-center">` elemen belül vannak rendezve, ami egy rugalmas sort hoz létre, amely középre igazított elemeket tartalmaz, az elemek között pedig 4 egységnyi (gy-4) függőleges távolság van.

- Minden kép egy külön `<div class="col-xl-3 col-lg-4 col-md-6">` elemen belül van elhelyezve, ami azt jelzi, hogy egy adott kép a 3/4/6-os oszlop méretének megfelelő helyet foglal el a galériában különböző képernyőméretek esetén.

- A `<img>` elemek tartalmazzák a képek elérési útját (`src` attribútum), az alternatív szöveget (`alt` attribútum) és a képek címét (`title` attribútum).

- A képek címei (általában az adott csapat neve és a stadion elnevezése) megjelennek, amikor a felhasználó az egerét a kép fölé viszi.

Ez a kód egy dinamikus galériát hoz létre, amely könnyen kezelhető és áttekinthető módon mutatja be a különböző stadionokat. A rugalmas elrendezésnek köszönhetően a képek megfelelően skálázódnak a különböző eszközök és képernyőméretek között.

Ez a kód egy videógalériát valósít meg a focimérkőzésekről készült videók megjelenítésére a weboldalon.

**HTML Tartalom:**

* + A <h2> címsor meghatározza a videógaléria címét: "Videók a focimérkőzésekről".
  + Két <div> elem, az egyik az időzített lapozáshoz (id="videoGallery"), a másik a lapozó gombok megjelenítéséhez (id="pagination").

**JavaScript Kód:**

* + A videoData tömbben tárolt adatok a videók elérési útját és típusát tartalmazzák.
  + Az videosPerPage változó meghatározza, hogy hány videót kell megjeleníteni egy oldalon.
  + Az autoPaginate függvény automatikusan vált az oldalak között egy megadott időközönként, így folyamatosan új videókat mutat.
  + Az autoPaginateInterval időzített intervallumot hoz létre az automatikus lapozáshoz.
  + Az onbeforeunload eseményfigyelő arra szolgál, hogy amikor az oldal bezárul, vagy a felhasználó elnavigál róla, leállítsa az automatikus lapozást.

Ez a kód egy rugalmas és interaktív módon megjeleníthető videógalériát hoz létre, amely automatikusan lapoz a felhasználó számára a rendelkezésre álló videók között.



## **4.6. Bajnokságról:**

**HTML Tartalom:**

* + A <section> elem meghatározza a szakasz kezdetét és végét, és az id attribútummal azonosítja azt (id="magyarlabdarugas").
  + A <div> elemek segítségével a tartalom és a kép elrendezését határozzák meg a szakaszban.
  + A kép az <img> elem segítségével van beillesztve, és a src attribútumon keresztül hivatkozik a képfájlra, amely az "mlsz.png" nevet viseli.

**Stílus és Animáció:**

* + Az data-aos és data-aos-delay attribútumok animációk vezérlésére szolgálnak, amelyeket a AOS (Animate On Scroll) könyvtár biztosít. Ezek segítségével az elemek megjelenése animálható, amikor a felhasználó görget az oldalon.

**Szöveg és Kép:**

* + A szakaszban egy szöveges tartalom található, amely bemutatja, hogy az oldalon minden információt megtalálhat a magyar futballról.
  + A szöveg egy címet tartalmaz a <h1> elem segítségével.
  + A kép a Magyar Labdarúgó Szövetség (MLSZ) logóját mutatja be.

Ez a kód az oldal azon szakaszát mutatja, amely összefoglalja és bemutatja a weboldal látogatóinak a magyar labdarúgással kapcsolatos információkat és tartalmakat. Az animációk segítségével dinamizálja az oldal megjelenését és vonzza a felhasználók figyelmét.

Ez a kód egy olyan szakaszt határoz meg, amely bemutatja a szponzorok logóit egy weboldalon. Néhány fontos részlet:

**HTML Tartalom:**

* + A <section> elem meghatározza a szakasz kezdetét és végét, és az id attribútummal azonosítja azt (id="szponzorok").
  + A <div> elemek segítségével a logók elrendezését határozzák meg a szakaszban.
  + Az <img> elemek az egyes szponzorok logóit jelenítik meg, és a src attribútumon keresztül hivatkoznak a képfájlokra, amelyek a "kepek/szponzorok/" könyvtárban találhatók.
  + Az alt attribútum segítségével alternatív szöveget adunk meg a képekhez, ami akkor jelenik meg, ha a kép betöltése vagy megjelenítése problémába ütközik.

**Stílus és Animáció:**

* + Az id="szponzorok" osztály attribútummal rendelkező <section> elemhez a "section-bg" osztályt is hozzárendeljük, ami egy háttérszínt vagy háttérképet adhat a szakasznak.
  + Az data-aos attribútum segítségével az animációk vezérelhetőek, amikor a felhasználó görget az oldalon. Ebben az esetben a logók "zoom-in" animációt kapnak, amikor megjelennek a képernyőn.

**Szponzorok logói:**

* + A szakaszban több <div> elemet találunk, amelyek mindegyike egy szponzor logóját tartalmazza.
  + Minden szponzor logóját egy <img> elem jeleníti meg, amelynek az id attribútuma "szponzorlogok".
  + A logók az align-items-center és justify-content-center osztályok segítségével középre vannak igazítva mind függőlegesen, mind vízszintesen.

Ez a kód egy olyan szakaszt hoz létre, amely bemutatja a weboldal szponzorainak logóit. Az animációk segítségével élénkítik az oldal megjelenését, és vonzzák a felhasználók figyelmét a szponzorokra.



Ez a kód a napi kvíz részt határozza meg a weboldalon, ahol a felhasználók válaszolhatnak kérdésekre.

**HTML Tartalom:**

* + A <section> elem meghatározza a kvíz szakasz kezdetét és végét, és az id attribútummal azonosítja azt (id="kviz").
  + A <div class="section-title"> elem tartalmazza a kvíz címét.
  + A <div class="container"> elem tartalmazza a kvíz elemeit.
  + A controls <div> elem tartalmazza a kezdés és a következő kérdés gombokat.
  + A question-container <div> elem tartalmazza a kérdéseket és válaszlehetőségeket.
  + A question <div> elem tartalmazza a kérdést.
  + Az answer-buttons <div> elem tartalmazza a válaszlehetőségek gombokat.

**Stílus és Animáció:**

* + A section-bg osztály beállítása segítségével a szakasz háttérszínt vagy háttérképet kap.

**JavaScript funkciók:**

* + Az időzítők segítségével lehetőség van a kvíz interaktív kezelésére, azaz a kérdések és válaszlehetőségek megjelenítésére.
  + Az eseménykezelők segítségével lehetőség van a gombok kattintásának kezelésére, például a "Kezdés" és a "Következő kérdés" gombokra való kattintásra.



Ez a JavaScript kód felelős a kvíz működtetéséért a weboldalon. Néhány fontos részlet:

**Változók definiálása:**

* + A kód elején változókat definiál a gombokhoz, a kérdésekhez és válaszokhoz kapcsolódó elemekhez.

**Eseménykezelők hozzáadása:**

* + Az addEventListener segítségével hozzáadva vannak a gombokhoz eseménykezelők, például a "Kezdés" és a "Következő kérdés" gombokra való kattintásra.

**Függvények:**

* + A startGame függvény elindítja a játékot, elrejti a kezdő gombot, nullázza az aktuális kérdés indexét, megjeleníti a kérdéseket és a válaszlehetőségeket.
  + A showQuestion függvény megjeleníti az aktuális kérdést és válaszlehetőségeket.
  + A resetState függvény visszaállítja a kvíz állapotát, például eltávolítja az előző válaszokat és elrejti a "Következő kérdés" gombot.
  + A selectAnswer függvény kezeli a válaszok kiválasztását és az ezekhez tartozó eseményeket.
  + A setStatusClass függvény hozzáadja vagy eltávolítja a stílusosztályokat az elemekhez a helyes vagy helytelen válaszok megjelenítéséhez.
  + A clearStatusClass függvény eltávolítja a stílusosztályokat az elemekről.

**Kérdések és válaszok:**

* + A questions tömb tartalmazza a kérdéseket és válaszlehetőségeket.
  + Minden kérdés egy objektum, amelynek van egy question tulajdonsága (a kérdés szövege) és egy answers tömbje (a válaszlehetőségek).

Ez a kód lehetővé teszi a felhasználók számára, hogy részt vegyenek a kvízben, megválaszolva a kérdéseket, és követve a logikát, hogy a következő kérdés megjelenjen, amikor az előzőre válaszolva haladnak.

Ez a HTML kód egy táblázatot hoz létre az NB1 (Labdarúgó Nemzeti Bajnokság) tabellájának megjelenítéséhez a weboldalon. Néhány fontos részlet:

**Táblázat struktúrája:**

* + A táblázat a <table> taggel kezdődik és záródik.
  + A táblázat fejlécét a <thead> tag tartalmazza, a fejléc sorokat pedig a <tr> (sor) és <th> (fejléc cella) tagok alkotják.
  + A táblázat testét a <tbody> tag tartalmazza, a sorokat és cellákat ugyanúgy <tr> és <td> tagokkal definiálják.

**Adatok megjelenítése:**

* + Minden sor egy-egy csapatot reprezentál a bajnokságban.
  + A táblázatban minden oszlop egy adatot tartalmaz a csapatok teljesítményéről (például győzelmek száma, döntetlenek száma, vereségek száma, összpontszám stb.).
  + A képek a csapatok logóit jelképezik, amelyek a <img> tag segítségével kerülnek beillesztésre.

**Stílusok és osztályok:**

* + A section-bg osztály segítségével a táblázat háttere stílusosan formázott lesz.
  + Az active-row és kiesohely osztályokkal megjelölhetjük az aktív sorokat vagy azokat, amelyek a kieső helyeket jelképezik.

Ez a kód lehetővé teszi az NB1 tabellájának dinamikus megjelenítését a weboldalon, amely frissülhet az aktuális bajnokság állásának megfelelően.



Ez a HTML kód egy táblázatot hoz létre az NB1 (Labdarúgó Nemzeti Bajnokság) góllövőlistájának megjelenítéséhez a weboldalon. Néhány fontos részlet:

**Táblázat struktúrája:**

* + A táblázatot a <table> taggel kezdjük és zárjuk.
  + A táblázat fejlécét a <thead> tag tartalmazza, a fejléc sorokat pedig a <tr> (sor) és <th> (fejléc cella) tagok alkotják.
  + A táblázat testét a <tbody> tag tartalmazza, a sorokat és cellákat ugyanúgy <tr> és <td> tagokkal definiálják.

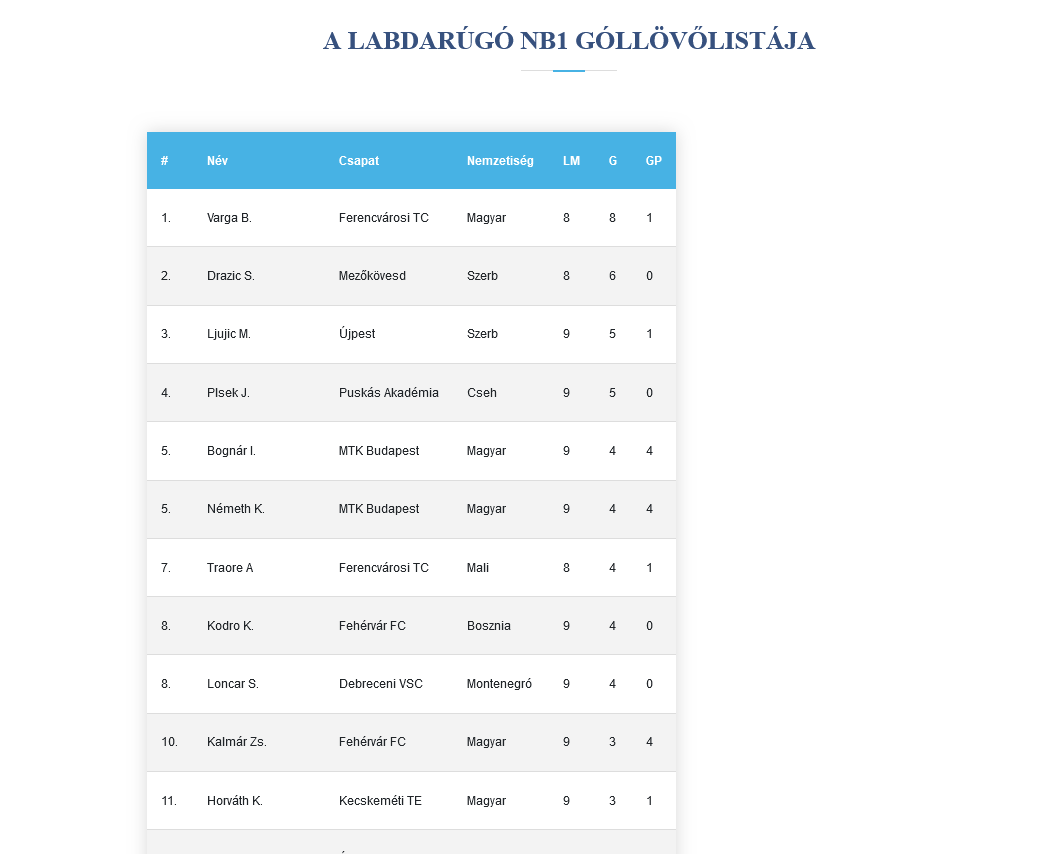
**Adatok megjelenítése:**

* + A táblázat minden sora egy-egy góllövő játékost reprezentál az NB1-ben.
  + A sorokban az adatokat oszlopokba rendezve jelenítik meg, például a játékos neve, csapata, nemzetisége, meccsek száma, lőtt gólok száma, góllövőpárbajok száma.

**Stílusok és osztályok:**

* + A táblázatot a styled-table osztály formázza stílusosan.
  + Az active-row osztály a kiemelt, aktív sort jelöli.

Ez a kód lehetővé teszi az NB1 góllövőlistájának dinamikus megjelenítését a weboldalon, amely szintén frissülhet az aktuális góllövők és gólszámaik szerint.



## **4.7.Eredmények:**

Ez a HTML kód egy olyan részt hoz létre, amely egy mérkőzés eredményét jeleníti meg az OTP bank liga adott fordulójából.

**Konténer struktúra:**

* + A mérkőzés megjelenítését egy fő konténer (<div class="match">) tartalmazza, amely a mérkőzés blokkját reprezentálja.

**Fejléc és tartalom:**

* + A mérkőzés fejlécében (<div class="match-header">) megjelenik a mérkőzés neve vagy típusa, például "OTP Bank Liga", egy kép segítségével.
  + A mérkőzés tartalmát (<div class="match-content">) a két csapat részletezi, az otthoni és az idegen csapat nevével és logójával, valamint a mérkőzés részleteivel, mint például a dátum és az idő, valamint az eredmény.

**Csapatok és eredmény:**

* + Minden csapatot egy-egy oszlop reprezentál, amelyben a csapat logója (<img>) és neve (<h2>) található.
  + Az eredmény megjelenítését egy külön oszlop tartalmazza (<div class="match-details">), amelyben a mérkőzés dátuma és időpontja, valamint az eredmény számai (<span>) láthatóak.

**Osztályok és stílusok:**

* + A különböző részeket és elemeket stílusosítják különböző osztályok, például match, match-header, match-content, column, team, team--home, team--away, team-logo, team-name, match-details, stb.

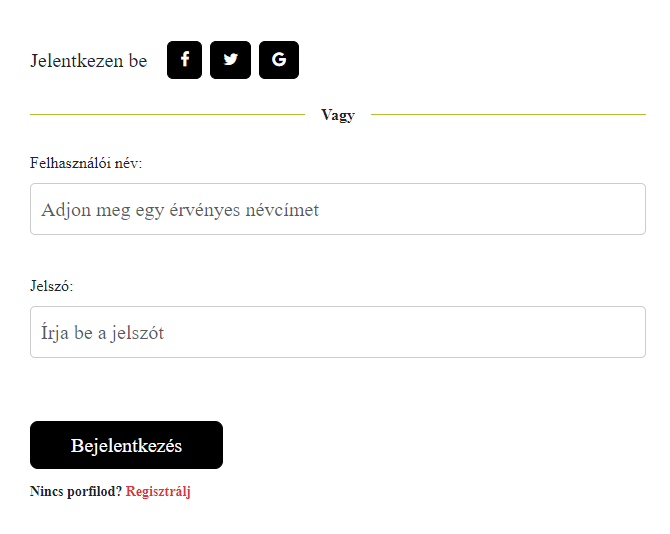
Ez a kód lehetővé teszi egy mérkőzés részletes megjelenítését az OTP Bank Liga adott fordulójából, beleértve a csapatokat, a dátumot és időt, valamint az eredményt.



# Tesztelés

|  |  |  |
| --- | --- | --- |
| Műveletek | Várt engedmény | Megjegyzés |
| bejelentkezés (jó adatok) | beléptet | ok |
| bejelentkezés (rossz) | hibaüzenet | ok |
| regisztráció (jó adatok) | regisztráció | ok |
| regisztráció (rossz adatok) | hibaüzenet | ok |
| módosítás (jó adatok) | módosítás | ok |
| módosítás (rossz adatok) | hibaüzenet | ok |
| törlés (jó adatok) | törlés | ok |
| törlés (rossz adatok) | hibaüzenet | ok |
| vélemény (jó adatok) | megosztás | ok |
| vélemény (rossz adatok) | hibaüzenet | ok |
| vásárlás (jó adatok) | vásárlás | ok |
| vásárlás (rossz válasz) | hibaüzenet | ok |
| kijelentkezés (jó adatok) | kijelentkezés | ok |
| kijelentkezés (rossz adatok) | hibaüzemet | ok |

## **5.1. Bejelentkezés:**

A bejelentkezés tervezés úgy lett létre hozva, hogy csak akkor engedi be a főoldalra, ha előtte regisztrál.

Amennyiben hibásan van beírva a név és jelszó, mint amikor regisztrált akkor nem engedi be az oldalra. Ha csak a nevet akarja beírni és a jelszót nem írja be a jelszót. Célszerű beírni mindent.

## **A képen szöveg, képernyőkép, Betűtípus, dokumentum látható Automatikusan generált leírás5.2. Regisztráció:**

Regisztráció készítést arra jutottam, hogy tudjanak regisztrálni adott mecsekre majd a mecssek alatt ki tudja választani, hogy milyen jegyet akarnak csoportosat vagy egyénit, ezek kiválasztás után a jegy mennyiségét és így tud kiválasztani majd a regiztráció után a főoldalra lépteti. Amit még meg tud adni a regisztrációban az igazolványszámot, email címét, amikor majd menni fog a mérkőzésekre akkor tudja majd igazolni magát, hogy tényleg az alapján lett regisztrálva a profilja és vásárolva a jegy. Ez a regisztrációs kép segítheti az elképzelés és a gondolkodást, hogyan lett elrendezve ez a felület és miért lett létre hozva. Ha csak a nevet és jelszó akar beírni a személy regisztrációhoz, nem engedi tovább, mert jelezni fog, hogy írja be a mezőbe a adatokat. Ha nem akarja a regisztráció folytatni ami elkezet akkor a reset gomb kattintásával tudja újra kezdeni a regisztrációját.

## **A képen szöveg, képernyőkép, Operációs rendszer, Betűtípus látható Automatikusan generált leírásA képen szöveg, képernyőkép, Betűtípus, szoftver látható Automatikusan generált leírás5.3. Módosítás:**

Az első kép a módosításnál az az adatot jeleniti meg, amivel a személy beregisztrált adatokkal és az tudja majd módosítani, ahogy a második képen is látszik. Ha azzal az adattal marad, amivel regisztrált azt is meg tudja vásárolni nem csak a módosított változatott. A felhasználó, amit tud módosítani az igazolványát, az email címét dátumot, a jegy mennyiségét és a jegy típusát („egyéni vagy csoportos”).

## **5.4. Törlés:**

A törlés azt a célt szolgálja, a jelenlegi bejelentkezett profilból automatikus törlésre kerül, ha rá kattint a gombra és kidobja a bejelentkezés felületre. És ott újra kell, majd kezdje a regisztrációval a profil létre hozását.

## **5.5. Vélemény:**

A képen szöveg, képernyőkép, névjegykártya, tervezés látható

Automatikusan generált leírásAz a célt szolgálja a felhasználónak, hogy tudjon írni a mérkőzésről véleményt, hogy milyen volt számára a mérkőzés jó vagy rossz. És írhat egy kicsi szövegrészt, meg csillagozhatja az adott mecsről. Az a cél volt, hogy ne csak a saját véleményét lássa a felhasználó egy másik oldalon, hanem más felhasználó véleményét is, hogy mit írtak és milyen volt az ők szemszögéből a mérkőzés. Fejlesztés alatt áll.

## **5.6. Vásárlás:**

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírásA vásárlással az volt a cél, hogy amikor regisztrált a felhasználó akkor azzal a regisztrált felhasználóbal tudjon vásárolni, ha módosítja az adatait nem azt adatai akkor az alapján tudja vásárolni jegyét. Az volt a cél, hogy kapjon egy üzenetet vagy egy emailt a telefonjára, hogy sikeres volt a vásárlás. Fejlesztés alatt áll.

## **5.7. Adattábla:**

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírásA képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírásEzek az adatbázisok a regisztráció, a vásárlás és a véleményadás MySQL adattáblája, ahova fel tudja tölteni a regisztráció adatai, a véleményét és a vásárláshoz az adatokkal. A regisztráció létrehozásával majd be tud lépni a saját profiljába és majd a fő oldalra töltődik be.

A vélemény tábla azért lett létre hozva, hogy feltudja töltetni a véleményét egy adott mecsről és egy másik oldalon jelenitse meg. Azért lett lássuk a felhasználó véleményét. Ha át megy erre a másik oldalra akkor mások véleményét is tudja majd megnézni meg a csillagozást ne csak a saját véleményét jeleníti meg, mert szerintünk az unalmas lehet, hogy csak a saját látszódik és másoké nem. Nyilván mindenkinek más lehet a vélemény nem mindenkinek lehet egyforma véleménye az adott mecsről. Fejlesztés alatt áll. A card („az az a vásárlás lenne”), ennél az adatbázisnál azt akartuk elérni, hogy a megadott regisztráció adataival tudjon vásárolni, ha nem módosítja, ha mégis módosítaná akkor az alapján is megtudja vásárolni a módosítás alapján. Még fejlesztés alatt áll a vásárlás.

# Értékelés, összefoglalás

## **6.1. Fejlesztési elérhetőségek:**

A fejlesztés során sok olyan hibába futottam/futottunk, mely komoly fejtörést és hosszú megoldáskeresgélést jelentett. A fejlesztés PHP, JS, CSS és MySQL segítségével lett létrehozva a project munkánk során. A projektünk, amin el lett készítve asztali számítógépen leginkább. Otthoni és iskolai számítógépjeink segítségével igyekeztünk a legjobb weboldalt létre hozni. Fejlesztés során a PHP dominál a legjobban. Mert könnyebb volt számunkra a felhasználói élményünk, és meghívni a különböző mappából a php létrehozott fájlokat. A kódolást, egy asztali alkalmazás Visual Studio Code alkalmazásával volt a domináns számunkra. Ez volt az egyik legjobb alkalmazás, amit jól tudtunk használni a PHP-hoz. A kódot, ahol tároltuk a github segítségével tudtuk igénybe venni, azért is mert több száz kódot hallottuk, hogy fel lehet feltölteni és alkalmasnak tartottuk ezt az egyik legjobb megoldásnak, hogy tároljuk a projektünket. Azért is vettünk igénybe, mert a meglévő kódulásunkat tudtuk fejleszteni ott, ahol abba hagytuk és így nem kellett mindig újra kezdeni az adott részt. A projektet piaci szintre jövőben még szóba jöhet fejlesztése, hogy kialakítsunk a megfelelő piaci formátumba. Még sok elképzelés lehet a projektben csak foglalkozni kell vele. A weblap egy jó alapanyag lehet egy jövőben induló vállalkozás megalapozásához. Könnyet átalakítható a weblap más online platformokra is, nem csak foci témában lehet elképzelni a jegyvásárlást például. Hanem egyéb weboldalakat is el lehet képzelni, mint például pizza vásárláshoz, online asztali pc-s játékokra el lehet gondolkodni. A fejlesztés a csillagos határ lehet az elképzelés, nyilván a hibák leszámítva majdnem tökéletes ez a weboldal, amit elkészítettünk.

## **6.2. Problémák a fejlesztés folyamán:**

A fejlesztés során, ami meg aggasztott bennünket, a kijelentkezés, nem tudtunk rá jönni, miért nem akarja a felhasználó profilt ki jelentkeztetni. Nagy nehezen megtaláltuk a megoldást és működő képes a kijelentkezés. A második probléma a módosítás volt, ami olyan volt, hogy nem akarta megjeleníteni az adatbázisból az adatokat csak hibát jelzet semmit mást. Ez a probléma már azóta megy és minden rendben működik. A harmadik probléma, amit nehezen tudtunk megoldani a vélemény írás a mecsekről. Ezt úgy oldottuk meg, hogy egy másik oldalon jelenítse meg a véleményeket. Ne csak a saját véleményét jelenítse meg, hanem több felhasználó véleményét is. A negyedik probléma, amit nehezen tudtuk megoldanunk a vásárlást volt. Az adatbázissal és a vásárlás php kódrészlet összeköttetésben vannak és az a cél lett volna, hogy a kártya adataival tudjon a felhasználó megvásárolni a jegyet, amikor regisztrált és egy adatlapon megjelenik. Van egy módosítás adatlapon, amin megjelenik a regisztrált adatai. Ha módosította az adatokkal az alapján meg vásárolhatja, ha mégis módosítaná az adatlapot akkor azzal a módosítással meg tudja vásárolni ajegyét. A formázással is sokat kellett foglalkozunk, hogy megfelelően jelenjenek meg a méretek, a képek, a szövegek, a menü sor, a módosítás stb. Nem volt ez se egyszerű, amikor azonos volt a meghívót id és class mert a kódolásunk különbözőek voltak és így elcsúsztak egymástól a css formázások. Megpróbáltuk megoldani, hogy egyes id-k és class-k ne különbözőnek egymástól, hanem egységessé próbáltuk megoldani az id és class hívásokat.