BGSzC Pestszentlőrinci Közgazdasági és Informatikai Szakgimnáziuma

1184 Budapest Hengersor 34.

1. Záró dolgozat

Küzdő Kert

Konzulens tanár:

Dobrocsi Róbertné Csilla

Készítette:Pekny Márk,Mózer Richárd,Rapcsák Marcell

Tartalom

[1 Bevezetés 3](#_Toc85723173)

[1.1 Feladat leírás 3](#_Toc85723174)

[1.2 A felhasznált ismeretek 3](#_Toc85723175)

[1.3 A felhasznált szoftverek 3](#_Toc85723176)

[2 Felhasználói dokumentáció 4](#_Toc85723177)

[2.1 A program általános specifikációja 4](#_Toc85723178)

[2.2 Rendszerkövetelmények 4](#_Toc85723179)

[2.2.1 Hardver követelmények 4](#_Toc85723180)

[2.2.2 Szoftver követelmények 4](#_Toc85723181)

[2.3 3. A program telepítése 4](#_Toc85723182)

[2.4 A program használatának a részletes leírása 5](#_Toc85723183)

[3 Fejlesztői dokumentáció 6](#_Toc85723184)

[3.1 Az alkalmazott fejlesztői eszközök 6](#_Toc85723185)

[3.2 Adatmodell leírása 6](#_Toc85723186)

[3.3 Részletes feladatspecifikáció, algoritmusok 6](#_Toc85723187)

[3.4 Tesztelési dokumentáció 7](#_Toc85723188)

[4 Összefoglalás 8](#_Toc85723189)

[4.1 Önértékelés 8](#_Toc85723190)

[4.2 Továbbfejlesztési lehetőségek 8](#_Toc85723191)

[5 Felhasznált irodalom 9](#_Toc85723192)

[6 Ábrajegyzék 10](#_Toc85723193)

# Bevezetés

A "Küzdősportok Közösségi Platformja" egy webalkalmazás, amelyet a küzdősportok iránt érdeklődő közösség számára fejlesztettünk. A célunk egy olyan platform létrehozása volt, ahol a felhasználók edzéseket, eseményeket, ranglistákat követhetnek nyomon, és élő közvetítéseket nézhetnek amiket az edzők közvetítenek. A program React alapú frontenddel, Node.js alapú backenddel, MySQL adatbázissal. A dokumentáció célja, hogy részletesen bemutassa a program telepítését, használatát, fejlesztési folyamatát, valamint a tesztelési és továbbfejlesztési lehetőségeket.

## Feladat leírás

A felhasználói dokumentáció célja, hogy a "Küzdősportok Közösségi Platformja" alkalmazás telepítésével és használatával kapcsolatos minden információt biztosítson a felhasználók számára. A dokumentáció segít a program kezelésének elsajátításában, a rendszerkövetelmények megértésében, a telepítési folyamat követésében, valamint a program funkcióinak részletes megismerésében. A fejlesztői dokumentáció pedig a program logikájának és kódjának megértését, valamint a további fejlesztést támogatja.

## A felhasznált ismeretek

A program fejlesztése során a következő ismereteket alkalmaztuk:

Frontend fejlesztés: React (komponensek, állapotkezelés, React Router), JavaScript, HTML5, CSS3 (Tailwind CSS).

Backend fejlesztés: Node.js, Express (REST API), MySQL (adatbázis-kezelés).

Adatbázis-tervezés: MySQL Workbench, relációs adatbázisok, SQL lekérdezések.

Verziókezelés: Git, GitHub.

Fejlesztési eszközök: Visual Studio Code, Postman (API tesztelés).

Tervezési ismeretek: UML diagramok (osztálydiagram, aktivitás-diagram), adatmodellezés.

## A felhasznált szoftverek

Visual Studio Code: Kódszerkesztő a frontend és backend fejlesztéséhez.

Node.js (v18.x): Backend futtatókörnyezet.

MySQL (v8.x): Adatbázis-kezelő rendszer.

MySQL Workbench: Adatbázis-tervezés és -kezelés.

Git: Verziókezelés.

Postman: API tesztelés.

React (v18.x): Frontend keretrendszer.

Vite: Build eszköz a React alkalmazáshoz.

Tailwind CSS: Stílusok.

SweetAlert2: Felhasználói értesítések.

React Router: Útvonalkezelés.

React Chart.js: Diagramok (pl. DashBoard).

Framer Motion: Animációk (opcionális, később eltávolítva)

# Felhasználói dokumentáció

## A program általános specifikációja

A "Küzdősportok Közösségi Platformja" egy webalkalmazás, amely a következő funkciókat kínálja:

Edzésnapló: A felhasználók nyomon követhetik, mely edzésekre jelentkeztek, és lemondhatják azokat.Nyomon követheted hogy hány edzésen vettél részt és motivációs szöveget is kapsz.

Ranglista: Az edzők rangsorának megtekintése hozzá adott edzések alapján.

Livestream: Élő közvetítések megtekintése. A program modern, reszponzív dizájnnal rendelkezik, és alkalmas közösségi interakciók támogatására.

Profil: A profil oldalon módosíthatod a saját adataidat, és beállíthatod, hogy milyen típusú edzések érdekelnek téged. Így személyre szabott edzéseket láthatsz.

Események: Itt láthatod milyen közelgő versenyek lesznek részletes leírást róla(Milyen fajta küzdő sportban lesz kik vehetnek részt és hogy hol lesz a verseny amit google maps segítségével is biztosítunk).

## Rendszerkövetelmények

### Hardver követelmények

Minimális konfiguráció:

Processzor: 1 GHz vagy gyorsabb.

RAM: 2 GB.

Tárhely: 500 MB szabad hely.

Internetkapcsolat: Stabil kapcsolat (legalább 1 Mbps).

Ajánlott konfiguráció:

Processzor: 2 GHz vagy gyorsabb.

RAM: 4 GB vagy több.

Tárhely: 1 GB szabad hely.

Internetkapcsolat: 5 Mbps vagy gyorsabb.

### Szoftver követelmények

Operációs rendszer:

Windows 10/11, macOS 10.15 vagy újabb, Linux (Ubuntu 20.04 vagy újabb).

Böngésző:

Google Chrome, Mozilla Firefox, Microsoft Edge (legfrissebb verzió).

Szükséges szoftverek:

Node.js (v18.x vagy újabb).

MySQL (v8.x vagy újabb).

Git (verziókezeléshez).

Adatbázis:

MySQL adatbázis (kuzdosportok nevű adatbázis).

## 3. A program telepítése

1. lépés: Előfeltételek telepítése

Telepítsd a Node.js-t a hivatalos weboldalról (nodejs.org).

Telepítsd a MySQL-t, és állítsd be a root felhasználót (jelszó nélkül, vagy állítsd vissza a jelszót, lásd korábbi üzenetemet).

Telepítsd a Githubot a fájl elérése érdekében ([GitHub](https://github.com/)).

Majd a Github Desktopot ahova clonezod majd a programot([Download GitHub Desktop | GitHub Desktop](https://desktop.github.com/download/)).

2. lépés: Projekt klónozása

Nyiss egy terminált, és klónozd a projektet:

git clone https://github.com/zaromunka\_kuzdosportok/React/KuzdoKert.git

Lépj be a projekt mappájába: cd KuzdoKert

3. lépés: Adatbázis beállítása

Hozd létre a kuzdosportok adatbázist a MySQL Workbench-ben.

Importáld az adatbázist a mellékelt josql8 fájlból:

mysql -u root -p kuzdosportok < josql8

4. lépés: Backend telepítése

Lépj a Backend mappába: cd Backend

Telepítsd a függőségeket: npm install

Indítsd el a szervert: npm run dev

5. lépés: Frontend telepítése:

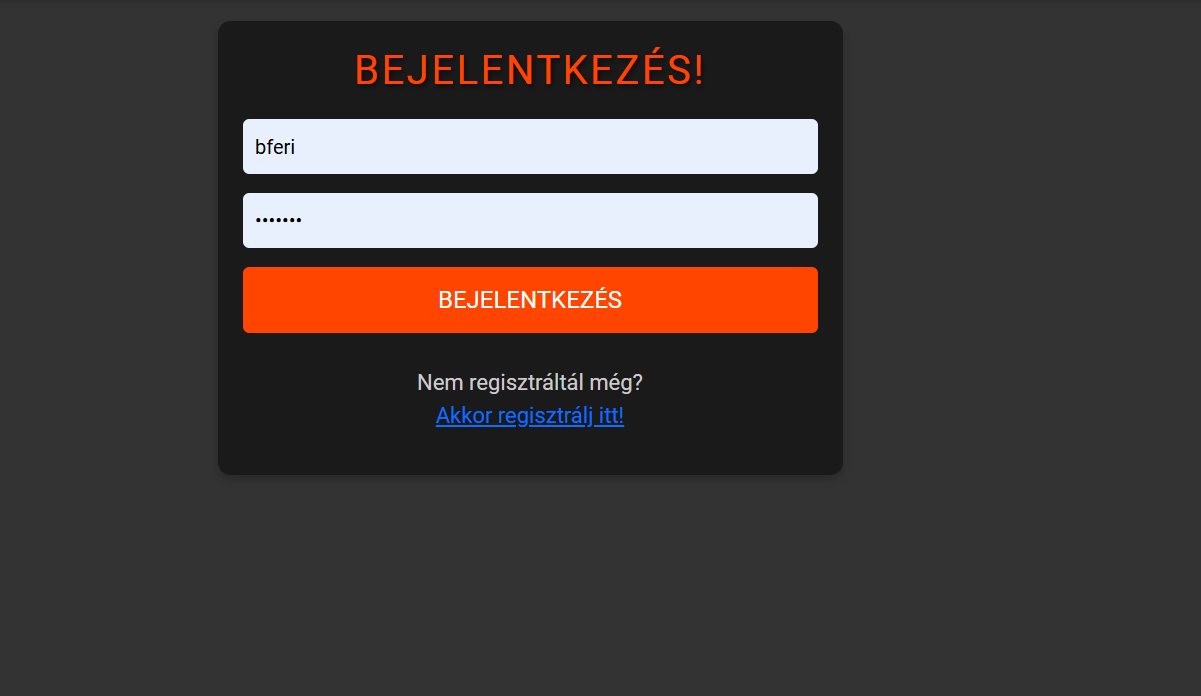
Lépj a Frontend mappába:cd Frontend

Telepítsd a függőségeket: npm install

Indítsd el a Vite szervert: npm run dev.

6. lépés: Program futtatása

Nyisd meg a böngészőt, és navigálj a http://localhost:5173 címre.

Jelentkezz be egy felhasználóval (pl. kpeter/bferi).  


## A program használatának a részletes leírása

Regisztrációnál kell megadni, hogy edzőként vagy látogatóként vagy jelen a weboldalon. Ezt követi a bejelentkezés. Amennyiben edzőként regisztrál az illető edzői oldalt talál az illető, ahol hozzátudja adni a saját klubját és az edzéseit a weboldalhoz. A látogató az edzői oldal helyett edzésnaplót talál, ahol azokat az edzéseket látja amikre már jelentkezett, valamint a kitűzött célját és az elért eredményét és még két motivációs szöveget. A sportok fülnél kiválasztott sportágnál meglévő klubbokat értékelni tudja csillagozással és szöveges értékeléssel egyaránt

# Fejlesztői dokumentáció

A fejlesztői dokumentáció célja, hogy a segítse program logikájának, illetve a program kódjának a megértését, illetve a program továbbfejlesztését. Szakemberek számára készül, elvárás tehát a pontosság és a szakmai jellegű stílus.

## Az alkalmazott fejlesztői eszközök

Fel kell sorolnod az összes olyan szoftver eszközt amelyet a program fejlesztéséhez, illetve a dokumentáció készítéséhez felhasználtál. (Pl. programozási nyelv, fejlesztői környezet, adatbázis-kezelő rendszer, kép-, szöveg-, zeneszerkesztő program stb.)

* Ha használtál mások által kifejlesztett modulokat, akkor azt is le kell írnod, a forrás pontos megjelölésével.
* Tájékozódj ezeknek a moduloknak a jogszerű felhasználásáról!

Ajánlott terjedelem: ½ -1 oldal.

## Adatmodell leírása

A feladatban alkalmazott adatbázis vagy adatszerkezet részletes leírása

* Adatbázis esetén az adattáblák leírása, a közöttük lévő kapcsolatok megadása, lehetőleg diagram is legyen
* Ha nincs adatbázis, akkor a program fő adatszerkezeteinek a specifikációja, célszerű diagramot is rajzolni
* OOP jellegű megvalósítás esetében az UML osztálydiagram

Ajánlott terjedelem: a feladat jellegétől függően 2-5 oldal.

## Részletes feladatspecifikáció, algoritmusok

A program lényeges függvényeinek, az osztályok metódusainak a specifikációja (mit valósít meg az adott függvény, illetve metódus, milyen paraméterei vannak, mi a visszatérési érték)  Az algoritmizálható részek leírása valamilyen algoritmus-leíró eszközzel (struktogram, pszeudo-kód, esetleg UML aktivitás-diagram) Ajánlott terjedelem: a feladat jellegétől függően 2-5 oldal. 5. Forráskód  A teljes forráskódot a nyomtatott dokumentációba nem kell beletenni!  Lehet viszont a nyomtatott dokumentációban a fontosabb kódrészeket magyarázattal szerepeltetni

## Tesztelési dokumentáció

1. Legalább 3 különböző teszteset részletes bemutatása.

* különböző felhasználó tevékenységek esetén hogyan reagált a program
* milyen üzeneteket kaptunk
* mi a teendő az egyes üzenetek esetében

1. Normál teszteset, extrém teszteset (bolondbiztosság tesztelése)
2. A tesztelés során kiderült hibák felsorolása A tesztelési dokumentációból derüljön ki, hogy ismered a különböző tesztelési módszereket (pl. fekete doboz, fehér doboz módszer)

Ajánlott terjedelem: a feladat jellegétől függően 2-5 oldal.

# Összefoglalás

A "Küzdősportok Közösségi Platformja" projekt sikeresen megvalósult, és a kitűzött célokat nagyrészt elértük. A platform lehetővé teszi a felhasználók számára, hogy edzéseket, eseményeket és ranglistákat kövessenek nyomon, valamint élő közvetítéseket nézzenek. A fejlesztés során számos kihívással szembesültünk, például a hírek képeinek betöltésével és a reszponzív dizájn biztosításával, de ezeket a problémákat megoldottuk. A projekt során szerzett tapasztalataink jelentősen hozzájárultak szakmai fejlődésünkhöz.

## Önértékelés

**Mózer Richárd**: A projekt során a kitűzött célok nagy részét sikerült elérnünk, de voltak olyan területek, ahol kompromisszumokat kellett kötnünk az időkorlátok miatt. Az alábbiakban összefoglalom a tapasztalatainkat, a felmerült problémákat és azok megoldását, valamint a saját fejlődésünket.

**Pekny Márk:** Az első félévhez képest 180 fokot fordult minden. Az elején nehezen kezdtünk bele és nehéz volt rávenni magam a tanulásra, aztán mire észbe kaptunk már alig volt hátra 4-5 hónapunk a munka befejezésére. A zárómunkánkba belefektetett rengeteg idő meghozta az eredményét. Ebben a pár hónapban többet tanultam, mint az elmúlt években bármikor. Az alapos megtervezés után kemény munkával de elértük azt, hogy úgy nézzen ki a weboldalunk, ahogyan mi azt szeretnénk és még rengeteg további ötletem van, hogy hogyan tudnánk még tovább fejleszteni és szándékunkban is áll a vizsga után még foglalkozni vele. Legtöbbet a weboldal frontend részével bajlódtunk, mert ott voltak a legnehezebb kihívások, mivel React-ban csináltuk így rengeteg új ismerettel gazdagodtunk, de végül, előbb vagy utóbb sikerült leküzdenünk minden akadályt. Röviden és tömören elégedett vagyok magammal és a csapattal is és nagyon örülök, hogy idáig eljutottunk. Nagyon élveztük az egész ide vezető utat.  **Rapcsák Marcell:**

### Kitűzött célok elérése

* **Edzésnapló és ranglista**: Az edzésnapló és a ranglista funkciók teljes mértékben működnek, a ranglistát modern dizájnnal és szűrési lehetőségekkel bővítettük.
* **Hírek szekció**: A hírek betöltése és magyarra fordítása sikeresen megvalósult, de a képek betöltése kezdetben problémát okozott. A proxy nélküli megoldás visszaállítása és a hibakezelés megoldotta a problémát.
* **Reszponzív dizájn**: Az oldal reszponzív lett, minden eszközön jól működik, de a kisebb képernyőkön a táblázatok görgethetősége további finomhangolást igényelhet.

### Felmerült problémák és megoldásuk

* **Képek betöltése**: A News API mezőjének betöltése CORS problémák miatt nem működött. Kezdetben proxy-t használtunk, de a felhasználói kérés alapján visszaállítottuk a közvetlen betöltést, és placeholder képekkel biztosítottuk a hibakezelést.
* **Világos/sötét mód eltávolítása**: A téma váltás logikáját el kellett távolítani, ami érintette az App.jsx, Home.jsx és hirek.css fájlokat. A folyamat során biztosítottuk, hogy a dizájn konzisztens maradjon.
* **Ranglista teljesítménye**: A ranglista oldalon a valós idejű frissítés túl gyakori volt, ezért a polling intervallumot 60 másodpercre növeltük.

### Saját fejlődés

A projekt során rengeteget tanultunk mind technikai, mind csapatmunkában szerzett készségek terén:

* **Technikai készségek**: Megtanultuk a React állapotkezelését, a REST API-k integrálását (News API, Google Translate API), valamint a reszponzív dizájn készítését Tailwind CSS segítségével. A MySQL adatbázis kezelését és az SQL lekérdezések optimalizálását is gyakoroltuk.
* **Problémamegoldás**: A képek betöltésével kapcsolatos problémák megoldása során megtanultuk, hogyan kell hibakeresést végezni (pl. böngésző fejlesztői eszközeivel), és hogyan kell alternatív megoldásokat találni (pl. placeholder képek használata).
* **Csapatmunka**: A projekt során szorosan együttműködtünk, és megtanultuk, hogyan osszuk fel a feladatokat (pl. Rapcsák a backend és adatbázis(dokumentáció) fejlesztésére,Pekny Márk a frontend dizájnra,PPT,(dokumentáció) Mózer Richárd Frontend elkészítése és az API integrációkra koncentrált illetve a dokumentáció ami közösen készült). A kommunikáció és a GitHub használata jelentősen javította a csapatmunkánkat és az otthoni munkát pedig a discord segítette.

Összességében a projekt során szerzett tapasztalatok megerősítették a programozási készségeinket, és felkészítettek minket a jövőbeli kihívásokra. Büszkék vagyunk arra, hogy egy működő, felhasználóbarát platformot hoztunk létre, amely valódi értéket nyújt a küzdősportok közösségének.

## Továbbfejlesztési lehetőségek

### Nem megvalósult ötletek

* **Valós idejű csevegés**: Terveztük egy chat funkció implementálását, amely lehetővé tette volna a felhasználók közötti valós idejű kommunikációt (pl. Socket.IO használatával). Az időkorlátok miatt ez nem valósult meg.
* **Értesítési rendszer**: Szerettünk volna egy értesítési rendszert, amely e-mailben vagy a felületen értesíti a felhasználókat az új edzésekről vagy eseményekről. Ez a funkció szintén kimaradt.

### Jövőbeli ötletek

* **Felhasználói kommentek a hírekhez**: A hírek szekcióban a felhasználók kommentelhetnének, ami növelné a közösségi interakciót.
* **Edzések részletesebb statisztikái**: Az edzésnaplóban részletesebb statisztikákat lehetne megjeleníteni (pl. havi bontás, edzéstípusok szerinti eloszlás).
* **Többnyelvű támogatás**: A platformot több nyelvre is le lehetne fordítani, hogy nemzetközi közönséget is elérjen.
* **Mobilalkalmazás**: A webalkalmazásból egy mobilalkalmazást lehetne készíteni React Native segítségével, hogy a felhasználók könnyebben hozzáférjenek a platformhoz.

# Felhasznált irodalom

 **Weboldalak**:

* React dokumentáció: <https://reactjs.org/docs/getting-started.html> (utoljára megtekintve: 2025.03.14.).
* Tailwind CSS dokumentáció: <https://tailwindcss.com/docs> (utoljára megtekintve: 2025.03.14.).
* News API dokumentáció: <https://newsapi.org/docs> (utoljára megtekintve: 2025.03.14.).
* Google Translate API dokumentáció: <https://cloud.google.com/translate/docs> (utoljára megtekintve: 2025.03.14.).

 **Elektronikus dokumentum**:

* Node.js hivatalos dokumentáció: <https://nodejs.org/dist/latest-v18.x/docs/api/> (letöltve: 2025.03.10.).
* AI (Chatgpt)

# Ábrajegyzék

[1. kép Ez a módosító ablak a programban 5](#_Toc63250394)