**E-sport Szervezési Felület**

**Dokumentáció**

**A projekt neve:** E-sport Versenykezelő Rendszer

**Fejlesztők:** Korcsok Roland és Gedovics Krisztián(14/D)

**A projekt ismertetője**

A mai világban az E-sport, mint sportágazat, egyre népszerűbbé válik. Leginkább a szórakozás és a versengés jellemzi.

Célunk, hogy egy olyan felületet nyújtsunk mindenki számára, amellyel rövid idő alatt és egyszerűen képesek házi bajnokságokat, vagy kisebb-nagyobb versenyeket szervezni iskolai szinten. A szervezés és a megvalósítás egy weboldalra lesz fejlesztve.

A weboldalunkon lehetőséget biztosítunk az események és a versenyek részletes megismerésére és nyomon követésére (mint például: az induló csapatok, játékosok, verseny legfontosabb információi stb.). Az érdeklődők bármikor regisztrálhatnak saját fiókkal, és jelentkezhetnek a versenyekre egyéni vagy csapat formációban, játéktól és játékmódjaiktól függően. Lehetőség nyílik a régebbi meccsek eredményeit visszanézni.

A szervezők egyszerűen és gyorsan hirdethetnek meg különböző játékokat, hogy ne kelljen több felületen bajlódni a regisztrációval és az események/versenyek lebonyolításával. A regisztrációs folyamat konzisztens és átlátható, így a felhasználók könnyen eligazodhatnak benne. A weboldalon szeretnénk egyszerűsíteni a versenyek lebonyolítását. Felügyelhetik, menedzselhetik az eseményekkel és a versenyekkel kapcsolatos tudnivalókat, amelyekhez a szervezők férnek csak hozzá. Így lehetőség nyílik számukra összesítve látni a jelentkezőket és az adataikat, valamint a csapatokat. Továbbá innen tudják beállítani a jelentkezés kezdetét és végét.

Ez a dokumentáció tartalmazza a projekttel kapcsolatos **technológiai** és **funkcionális** ismertetőjét, emellett a **telepítési útmutatót**.

**A projekt funkciói**

1. **Szervezői felület:**

A regisztrált szervezők eseményeket és azokon belül versenyeket hozhatnak létre. Ezekhez különböző attribútumokat állíthatnak be (pl. jelentkezés kezdete, vége, verseny kezdete, vége, helyszín, név, kép). A szervezők kezelik a jelentkezéseket: elfogadhatják vagy elutasíthatják azokat. Minden szervező csak a saját maga által létrehozott eseményeket, versenyeket és meccseket szerkesztheti. A jelentkezés lezárultával meccseket szervezhet, azaz kijelölheti, mely csapatok játszanak egymással. A szervezők bannolhatnak felhasználókat vagy feloldhatják a tiltást. Lehetőségük van jelszó visszaállítást kérni emailen keresztül, valamint saját adataikat is módosíthatják.

1. **Felhasználói felület:**

A regisztráció után a felhasználó megerősítő emailt kap, amely rendszerint a spam mappába kerül. A megerősítés után a regisztráció végleges. A felhasználó belépve szerkesztheti az adatait, csapatot hozhat létre, módosíthatja vagy törölheti azt, új csapatkapitányt jelölhet ki. Csapatkapitányként más játékosokat hívhat meg csapatába. A felhasználó láthatja a saját csapatait és azokat is, amelyekben csak tagként van jelen. Lehetőség van versenyre jelentkezni, csapatonként legfeljebb 7 játékossal. A jelentkezéskor kiválaszthatja, hogy mely játékosok indulnak az adott versenyen.

1. **Közös funkciók (szervezők és felhasználók számára):**

* Csapatok és versenyek szűrése név alapján.
* Események és játékosok szűrése név alapján.
* Játékos és szervezőprofil megtekintése.
* A főoldalon az iskola Twitch csatornájának beágyazott élő közvetítése megtekinthető.
* A főoldalon három esemény jelenik meg státusz szerinti sorrendben: aktív, még el nem kezdett, lezárult.

**Felhasznált technológiák**

1. **Adatbázis:**
   * + MySQL (MariaDB) – XAMPP
2. **Backend:**
   * + JavaScript – Node.JS
     + Express
     + Cors
     + Prisma ORM (Object-Relational Mapping)
     + Nodemailer (e-mail levelezés)
     + Bcryptjs (visszafejthetetlen hash titkosítás)
     + Multer (médiafájl feltöltés, jelenleg kép)
     + Node-cron (automatizált kódok futtatása)
     + Dotenv (környezeti változók)
     + HTTPS
     + Nodemon (fejlesztésnél automatikus program újraindítás)
     + Middleware használat (felhasználói autentikáció)
3. **Frontend:**
   * + JavaScript – Node.JS
     + React-Vite
     + React Router DOM
     + TailwindCSS
     + DaisyUI
     + React Toastify
     + React Icons
     + React Twitch Embed Video
4. **Fejlesztési eszközök:**
   * + Fejlesztői környezet: Visual Studio Code
     + Verziókezelés: GitHub

**Telepítési útmutató**

A projekt **GitHub**-ra van feltöltve, amelyet [ezen](https://github.com/RolandKorcsok/E_sport_verseny_szervezo_menedzselo_weboldal) a hivatkozáson érhet el.   
Töltse le „**.zip”** formában a projektet és csomagolja ki, vagy használja a **Git/GitHub Desktop** programokat a *repository* leklónozásához. Az útmutató további részei a projektben lévő mappákra és fájlokra vonatkoznak. A mappák és a fájlok kereséséhez, valamint a frontend és backend programok futtatásához ajánljuk a Visual Studio Code fejlesztői környezet használatát, így az alábbi útmutató ezt fogja alkalmazni.

1. **Adatbázis létrehozása:**
   * + Telepítse, majd nyissa meg a(z) **XAMPP**-t, azon belül az **Apache** és a **MySQL szervert** indítsa el.
     + Nyisson egy böngészőt, majd írja be a kezelőfelület eléréséhez:  
       **localhost/phpmyadmin** vagy **127.0.0.1/phpmyadmin**
     + A felületen kattintson az **SQL** fülre, majd a projekt mappában lévő „**Adatbázis**” mappában található   
       *„****esport\_database kényszerekkel.sql***” fájlban található első három sorát másolja ki és futtassa le. A fájlban lévő többi sor az adatbázis SQL kódját tartalmazza, amely a táblák és azok kulcsait készítik el. Ezekre nincs szüksége, mivel a következő részben fogjuk legenerálni az adatbázis tábláit.
2. **Backend és az adatbázis legenerálása:**
   * + A „**Project**” mappán belül a „**backend**”mappához indítson terminált (Parancssor és PowerShell egyaránt működik).
     + A **node modulok** letöltéséhez futtassa le az „**npm i**” parancsot.
     + A Prisma ORM végett az adatbázist a megfelelő parancsokkal le lehet generálni. Megtekinteni a sémafájlt a **prisma** mappában tudja, **schema.prisma** néven. Az adatbázist a „**.env**” környezeti változóból fogja elérni, amely a következő: „*mysql://root@localhost:3306/db\_esport”.*
     + Futtassa az „**npx prisma migrate dev**” parancsot, hogy a migráció megtörténjen. Adhat nevet a migrációnak, de ez nem kötelező. Az adatbázisban megjelentek a táblák, a kulcsok is beállításra kerültek.
     + A program futásához futtassa az „**npm start**” parancsot. A backend program a **8000-es porton** érhető el **localhost**-on, a program futásakor is kiírja. **(**[***https://localhost:8000***](https://localhost:8000)***)***
     + Mivel a program nem hivatalos SSL tanúsítványt tartalmaz a „**config**” mappában, ezért az **API kéréseket** csak akkor fogja rendesen lekezelni, ha a böngészőbe beírja a fenti **localhost címet**, majd a kockázati figyelmeztetésnél **továbblép**.  
       Az alábbi szöveget kell látnia: „Esport adatbázisos Backend”
3. **Frontend (Felhasználói és Szervezői felület):**
   * + A „**Project**” mappán belül található a „**frontend**” mappa, ahol több mappa is. Ezek közül az „**organizer**” és az „**user**” mappákat fogjuk igénybe venni.
     + Nyisson a két mappához terminálokat, majd futtassa le az „**npm i**” parancsot. Ha végzett a letöltés, akkor indítsa el a frontend programokat az „**npm run dev**” paranccsal.
     + **Figyelem: először a felhasználói felület frontend programát indítsa el, és azután a szervezőit! Ennek okát a backend mappában lévő környezeti változóiban található meg.  
       (E-mail küldésnél a megerősítő és a jelszó visszaállító levelek ezen változók szerint vannak elkülönítve.)**
     + A felhasználói felületek az **5173-mas**, a szervezői felület pedig az **5174-es portokon** futnak.
     + Az „**Adatbázis**” mappában megtalálhatók az előre megírt példasorok a „***peldasorok.sql***” fájlban. Ezek közül a „**Pictures**” tábla adatai a legfontosabbak, mivel azok tartalmazzák az alapképek elérési útvonalait, amelyeket letárolunk, és szükségesek a képek megjelenítéséhez, így annak a táblának a beszúró SQL kódját futtassa le!  
       Egyébként kimásolhatja az egész fájl tartalmát, és az adatbázis **SQL** fülén lefuttathatja.  
       Más további teendője nincs.