*Technikai Dokumentáció*

TradeBlazer

Aranyosi Dániel – Széll Ádám

Bevezetés

Üdvözöljük a TradeBlazer Technikai Dokumentációjában! Ez a dokumentum célja, hogy

segítséget nyújtson az alkalmazás használatához, és részletes iránymutatást kínáljon a

technikai részletekről.

Cél

A dokumentáció célja, hogy:

Ez a technikai dokumentáció a weboldal fejlesztői és karbantartói számára készült, hogy

megértse a rendszer működését és karbantarthatóságát.

Közönség

Ez a dokumentáció azoknak szól, akik az alkalmazást telepítik, karbantartják vagy fejlesztik.

A dokumentum segít mindazoknak, akik mélyebben meg akarják érteni az alkalmazás

működését.

Elérhetőség

Amennyiben bármilyen kérdése, észrevétele vagy támogatási igénye van a TradeBlazer

alkalmazással kapcsolatban, kérjük, vegye fel velünk a kapcsolatot a következő

elérhetőségeken:

• E-mail cím: aranyosi.daniel-2020@keri.mako.hu

• E-mail cím: szell.adam-2020@keri.mako.hu

• Szívesen állunk rendelkezésére, hogy segítsünk és válaszoljunk minden felmerülő

kérdésre vagy kérésekre.

Mappastruktúra

* /backend – Ez a mappa tartalmazza az összes kódot a backend-nek, beleértve a server fájlait, és az adatbázishoz szükséges fájlokat.
* /doc – Ez a mappa tartalmazza ezt a dokumentumot, meg az összes többi dokumentációt, issuekat és egyebeket
* /frontend – Ez a mappa tartalmazza a frontendhez az összes szükséges fájt
* node\_modules mappák – Ezek a mappák tartalmazzák a nodehoz szükséges csomagokat amiket használunk a projektünkben
* /frontend/public – Ezek azok a fájlok amikhez hozzá lehet férni gond nélkül, „public”
* /frontend/src/assets – Ez a mappa tartalmazza az egyéb összes nem-kód anyagot az oldalunkhoz, ide tartozik például a felhasználóknak az avatárjaik, vagy az összes css az oldalhoz
* /frontend/src/components – Ide mennek azok a vue komponensek amiket többször is használhatunk, és összerakják az oldalunkat szerkezetileg
* /frotend/src/router – Ez a mappa tartalmazza a vue routerünk fájlait
* /frotend/src/stores - Ez a mappa tartalmazza az összes store-unkat a weboldalunkhoz
* /frotend/src/utils - Ez a mappa tartalmazza az összes egyéb eszközt az oldalunkhoz, az írás pillanatában a validationhöz global functionoket
* /frotend/src/views – Ez a mappa tartalmazza az oldalunk, felhasználó által interaktált vue oldalait, általában egy route-al vannak összekötve

Api endpontok:

Minden api endpont a backend server címe alatt a /api/ alatt található

Public endpontok:

* /api/login – POST
* /api/register – POST
* /api/types – GET
* /api/health - GET

**User/privát endpontok:**

* /api/checkfile - POST
* /api/users/:id/avatar – DELETE/PUT
* /api/users/:id – DELETE/PUT
* /api/users – GET
* /api/data – GET
* /api/types – GET
* /api/trending – GET
* /api/portfolio/:userId - GET
* /api/portfolio/deposit – POST

**Webes technológiák:**

* **HTML**: A weboldalak alapvető szerkezetét biztosító jelölési nyelv.
* **CSS**: A weboldalak vizuális megjelenésének kialakítására szolgáló stílusleíró nyelv.
* **JavaScript**: Interaktív funkciók megvalósítására használt szkriptnyelv.
* **TypeScript**: A JavaScript egy erősen típusos változata, amely nagyobb biztonságot és jobb fejlesztési élményt biztosít.

**Adatbázis technológiák:**

* **SQL**: Relációs adatbázisok kezeléséhez használt lekérdezési nyelv.
* **phpMyAdmin**: Webalapú adatbázis-kezelő eszköz MySQL adatbázisokhoz.

**Szerver oldali programozás technológiái:**

* **Express.js**: A Node.js keretrendszere, amely egyszerűsíti a szerveroldali alkalmazások fejlesztését.
* **CORS**: Mechanizmus a kereszt-domain kérések kezelésére.
* **Nodemon**: Automatikusan újraindítja a Node.js alkalmazást kódváltozások esetén.
* **npm**: A Node.js csomagkezelője, amely segít a függőségek kezelésében.

**Frontend technológiák:**

* **Vue.js**: Progresszív JavaScript keretrendszer frontend alkalmazások fejlesztéséhez.
* **Pinia**: Vue.js állapotkezelő könyvtár, amely egyszerűbb és hatékonyabb adatkezelést tesz lehetővé.
* **Tailwind CSS**: Utility-first CSS keretrendszer, amely segít a gyors és hatékony stílusozásban.
* **FontAwesome**: Ikonkészlet a weboldalak vizuális elemeinek gazdagítására.
* **vue-file-upload**: Fájlfeltöltést segítő Vue.js komponens.
* **vue-advanced-cropper**: Képvágást lehetővé tevő Vue.js könyvtár.
* **vue-draggable-next**: Drag and drop funkcionalitást biztosító Vue.js könyvtár.
* **vue-router**: Útvonalkezelő Vue.js alkalmazásokhoz.
* **date-fns**: Dátum- és időkezelési könyvtár.
* **axios**: HTTP kérések kezelésére használt könyvtár.

**Fejlesztői környezet:**

* **Visual Studio Code**: Könnyen bővíthető, hatékony kódszerkesztő.
  + **Vue Extension**: Vue.js fejlesztési környezet kibővítésére.
  + **Tailwind CSS Intellisense Extension**: Tailwind CSS osztályok gyors keresését és alkalmazását segítő bővítés.
* **Git/Git Bash**: Verziókezelő rendszer kód verzionálására és együttműködésre.
* **XAMPP MySQL-el és phpMyAdminnal**: Lokális fejlesztési szerver MySQL adatbázissal.
* **Chrome/Firefox**: Webes alkalmazások tesztelésére használt böngészők.

Kapcsolat és Köszönetnyilvánítás

Köszönjük, hogy időt szánt a Tradeblazer technikai dokumentációjának áttekintésére! Bízunk benne, hogy az itt található információk hasznosak voltak, és megkönnyítik a fejlesztési és karbantartási feladatait.

Ha további kérdése merülne fel, vagy támogatásra lenne szüksége, forduljon hozzánk bizalommal – örömmel segítünk! Értékeljük az elkötelezettségét és érdeklődését, hiszen ezek kulcsfontosságúak a közös sikerhez.

Köszönettel,

Aranyosi Dániel & Széll Ádám