**ZÁRÓDOLGOZAT**

**Egersdorfer Dominik**

**5/13 SZOFT**

**BAJAI SZC TÜRR ISTVÁN TECHNIKUM**

**SZOFTVERFEJLESZTŐ**

**ZÁRÓDOLGOZAT**

Tier List

**Egersdorfer Dominik**

2024

**NYILATKOZAT A ZÁRÓDOLGOZATRÓL**

Alulírott (név) tanuló

**kijelentem, hogy**

című záródolgozatomat (nyomtatott és elektronikus formában) a Bajai SZC Türr István Technikumának pedagógusai és tanulói:

* + felhasználhatják (pl. hivatkozás alapjául, olvasótermi használatra) későbbi munkájukhoz a szerzői jogok tiszteletben tartása mellett).
  + nem használhatják fel (titoktartási nyilatkozat csatolása mellett).

Ugyanakkor kijelentem, hogy a záródolgozat ***saját munkám eredménye*.**

Baja, 2024. április 25.

aláírás

**Tartalomjegyzék**

1. Bevezetés 7

2. Felhasználói dokumentáció 8

2.1. Főoldal 8

2.1.1. Regisztráció 9

2.1.2. Bejelentkezés 10

2.2. Listák 10

2.2.1. Új lista 11

2.2.2. Jogosultságok 12

2.3. Lista készítő 12

2.3.1. Karakterek 14

2.3.2. Karakter hozzáadása 14

2.3.3. Karakter módosítás 15

2.3.4. Kategóriák 16

2.3.5. Kategória hozzáadása 16

2.3.6. Kategória módosítás 17

2.4. Profil 17

2.4.1. Statisztikák 18

2.4.2. Adatok módosítása 18

2.5. Email hitelesítés 20

2.6. Navigáció 21

3. Fejlesztői dokumentáció 22

3.1. Telepítés 22

3.1.1. Npm 22

3.1.2. Bun 22

3.1.3. Telepítési parancsok 22

3.2. Fejlesztés során használt szoftverek 23

3.2.1. Visual Studio Code (^1.88.0) 23

3.2.2. XAMPP (^3.3.0) 23

3.2.3. Git & Github 23

3.2.4. Atlassian Jira 23

3.2.5. Figma 24

3.2.6. Firefox (^124.0.2) 24

3.3. Frontend 24

3.3.1. Függőségek 24

3.3.2. Vite 25

3.3.3. React 25

3.3.4. Tailwind 25

3.3.5. dnd kit 25

3.3.6. FontAwesome 25

3.4. Backend 25

3.4.1. Függőségek 26

3.4.2. Környezeti változók (.env) 26

3.4.3. Adatbázis 27

3.4.4. Útvonalak 29

3.5. Fájlrendszer 30

3.5.1. Backend mappák: 31

3.5.2. Frontend mappák: 31

3.6. Továbbfejlesztési lehetőségek 32

3.6.1. Főoldal Demo 32

3.6.2. Beállítások 32

3.6.3. Téma mentes Tier List 32

3.6.4. Barátlista, csevegő 32

3.6.5. Elfelejtettem a jelszavam 32

4. Összegzés 33

5. Ábrajegyzék 34

# Bevezetés

A szakdolgozatom egy anime témára specializálódott Tier List weboldal, ahol karaktereket lehet rangsorolni.

A Tier List röviden egy olyan lista, amiben rangsorolni tudunk valamilyen elemeket (például szereplőket), valamilyen szempontok alapján (például erősség). A kiválasztott elemeket különböző kategóriákba, szintekbe, azaz „tier”-ekbe sorolhatjuk és ezekben általában felül szerepelnek a „jobb”, alul pedig a „rosszabb” elemek. A témát és az abban szereplő elemeket, szinteket mi választjuk, és azt is, hogy mi alapján szeretnénk elrendezni. Az én oldalamon a téma adott, az elemek pedig az anime karakterek, de a besorolási szempontok a felhasználó kreativitására vannak bízva.

A témaválasztásom azért erre esett, mivel az általam ismert oldalakon sok bővítési lehetőséget láttam és a legtöbb oldalon egy képen kívül semmilyen információt nem kaphatunk az adott elemekről. Kevés anime témára specializálódott oldalt találtam, ezért egy jól átgondolt, hasznos funkciókkal kibővített felület volt az elképzelés, ahol a felhasználó szabadon létrehozhatja saját listáját.

Az egyik leghasznosabb funkció az anime Tier List-ek elkészítéséhez, egy karakter kereső, ami a Jikan API segítségével egyszerűen megvalósíthattam. A cél az, hogy ha valaki tudja kit szeretne felhasználni listájában, akkor egyszerűen csak rákeres és már helyezheti is kedve szerint. Bármikor módosíthatja a már bent lévő karaktereket, vagy ha már valamiért nincs rá szüksége, akkor törölheti azokat. A kategóriákat is szabadon testre szabhatjuk, attól függően mi lesz a lista témája.

A közös listák készítése se okoz problémát, mivel minden – a lista oldalán tartózkodó – felhasználó látja az adott pillanatban történő módosításokat, tehát ha egy karakter áthelyezésre kerül, vagy valamilyen adata szerkesztve lett, akkor arról mindenki tudni fog. Az oldalon jogosultságokat lehet társítani a felhasználóknak a listáikhoz, hogy korlátozzuk vagy bővítsük az elvégezhető műveletek számát.

# Felhasználói dokumentáció

Az oldal Anime karakterekből álló Tier List-ek létrehozásának ad egy gyors és egyszerű felhasználói felületet, ahol szabadon elkészítheti a saját listáit.

A Tier List egy olyan listát jelent, amelyben adott témák karaktereit, szereplőit, elemeit rangsorolják valamilyen szempontok, például hatékonyság, hasznosság vagy erő alapján. Legtöbb esetben – mint ezen az oldalon is – szintekre („tier”-ekbe) sorolják az elemeket, ebben a legjobbak a legfelsőbb, míg a kevésbé „jók” az alsóbb kategóriákban helyezkednek el. A Tier List segít az embereknek eligazodni az adott témában, jelen esetben akár egy anime vagy animék közötti világokban, hogy ezáltal több információt szerezhessenek egy-egy karakterről. Ezen a területen több különböző gondolat is kialakulhat, így ezek kifejezhetnek véleményt is.

## Főoldal

A főoldalunkon találhatunk néhány információt az oldalról. Elérhetjük a regisztrációt és a bejelentkezést is.



1. ábra: Főoldal

Az oldal használatához kötelező a regisztráció, ez nélkül semmilyen funkciót nem érhetünk el.

### Regisztráció

Regisztráció során három adatot kell megadnunk:

* **Felhasználónév:** Bármilyen szöveg, ami a felhasználó neveként fog szolgálni az oldalon. Kétféle hibaüzenetet kaphatunk:  
   Üres mezőnél: „Adj meg egy felhasználónevet!”  
   Foglalt név esetén: „A megadott felhasználónév már foglalt!”
* **Email:** Egy létező email cím, ami segítségével könnyedén beazonosítható, hogy valóban tiéd a felhasználó. Háromféle hibaüzenetet kaphatunk:  
   Üres mezőnél: „Adj meg egy emailt!”  
   Hibás formátumú email címnél: „Az email cím nem megfelelő!”  
   Foglalt e-mail címnél: „A megadott email cím már foglalt!”
* **Jelszó:** Legalább 8 karakter hosszú szöveg, ajánlott számokat és speciális karaktereket is használni, de ezt a rendszer nem követeli meg. Kétféle hibaüzenetet kaphatunk:  
   Üres mezőnél: „Adj meg egy jelszót!”  
   Rövid jelszónál: „A jelszavadnak legalább 8 karakter hosszúnak kell lennie!”
* **Jelszó újra:** Ez az előző mezőben megadott jelszó megismétlése, csökkentve az elgépelés esélyét. Kétféle hibaüzenetet kaphatunk:  
   Üres mezőnél: „Adj meg egy jelszót!”  
   Ha a két jelszó ugyanaz: „A két jelszó nem egyezik!”

2. ábra: Regisztráció

### Bejelentkezés

Bejelentkezés során a már regisztrált felhasználónk nevét és jelszavát szükséges megadnunk. Az emailünkben kapott hitelesítést csak azután tudjuk elvégezni, miután belépett a weboldal felületére.

Sikeres belépés után az oldal azonnal átirányít a listáinkhoz.

* **Felhasználónév lehetséges hibaüzenetei:**

Üres mezőnél: „Adj megy egy felhasználónevet!”

Hibás felhasználónévnél: „A felhasználó nem található!”

* **Jelszó lehetséges hibaüzenetei:**

Üres mezőnél: „Adj meg egy jelszót!”

Helytelen jelszónál: „Hibás jelszó!”

3. ábra: Bejelentkezés

## Listák

Ez az oldal ad felületet, hogy a felhasználó a listákat kezelje, létrehozzon újakat, módosítsa azokat vagy akár törölhessen belőlük. Minden listához külön-külön kapcsolódnak jogosultságok, amiknél felhasználók kaphatnak engedélyt a kiválasztott lista elérésére.

Amennyiben már megnyitott valaki egy listát, a jobb alsó sarokban egy navigáció jelenik meg, ahol váltani lehet a jelenlegi és a lista készítő oldal között.

A bal oldali részen megtalálhatóak a már létrehozott listák, ezekre kattintva módosíthatja, megnyithatja vagy törölheti azokat.

A képen képernyőkép, szoftver, Multimédiás szoftver, Grafikai szoftver látható

Automatikusan generált leírás

4. ábra: Listák

Alapértelmezetten az új lista létrehozás és a lista kereső jelenik meg.

### Új lista

Egy listának az alábbi adatokat tudod megadni:

* **Név:** A lista megjelenő neve, ez alapján tudod beazonosítani. Ez egy kötelező mező.
* **Leírás:** Rövid információ. Nem kötelező mező.
* **Státusz:** Ezzel a funkcióval megjelölheted a jelenlegi állapotot, de hatása nincs a listádra.
* **Láhatóság:** A publikus listákra bárki rákereshet és megnézheti, a privátot kizárólag a jogosultsággal rendelkező felhasználók tekinthetik meg.

5. ábra: Új lista

Az új lista létrehozás alatt találhatunk egy lista böngészőt, amiknél láthatjuk az összes velünk megosztott és publikus listákat. Látható, hogy ki és mikor módosította utoljára az adott listát, valamint, hogy a tulajdonosa milyen státuszba sorolja azt.

Egy listára kattintva megtekinthetjük annak összes adatát és a már hozzáadott felhasználók jogosultságait.

### Jogosultságok

A jogosultságoknál az alábbi adatokat lehet megadni:

* **Felhasználó:** Ide a pontos felhasználónevet kell megadni.

**Lehetséges hibaüzenetek:**

Üres mező: Írj be egy felhasználónevet!

Helytelen felhasználónév: Nem található felhasználó!

* **Jogosultság:** A felhasználóknak három különböző jogosultságot lehet adni:

**Megtekintés:** Meg tudják tekinteni. Privát listánál van jelentősége.

**Mozgatás:** Karakterek és kategóriák mozgatása.

**Szerkesztés:** Lista tulajdonságain kívül minden elérhető számára.

6. ábra: Jogosultságok

Új jogosultságot a jobb oldalon fent található fehér ikonra kattintva tudunk hozzáadni, míg módosítani a már meglévő felhasználókra kattintva. Mindkettőnél ugyanott – alul – megjelenik az ehhez szükséges felület.

## Lista készítő

Minden lista módosításához erre az oldalra kell eljutnia a felhasználónak, amit a listák kezeléséből, az oldalt található navigációs menüből vagy a profil oldalról nyithat meg.

Itt mindenki a jogosultságaitól függően láthat gombokat és hajthat végre műveleteket. A „Kategória hozzáadás” és a karakter hozzáadást jelző gomb az alsó sávban nem látható, ha Szerkesztésnél gyengébb jogosultsággal rendelkezik.

A legalsó üres szint a nem használt karaktereknek ad helyet, ide tudunk felvenni újakat is. A felette található színes sávok már az általunk létrehozott kategóriák, amiket tudunk módosítani. Lista létrehozás után négy darab előre beállított szint kerül létrehozásra (A, B, C, D).

Az oldalon jelenlévő felhasználók azonnal láthatják egymás műveleteit, áthelyezett karaktereket, kategóriákat, módosított adatokat, törléseket. Ez nagyban elősegítheti a többszemélyes listakészítés végrehajtását vagy akár egy lista elkészülésének megfigyelését. A jelenlévő felhasználókat a lista címe alatt fogjuk látni.

Jobb felső sarokban két ikon található:

* ** Oldal újra töltése:** Ahhoz, hogy ne kelljen az egész oldalt újra betölteni, ezzel a gombbal kizárólag a megnyitott tier listet lehet frissíteni.
* ** Megosztás:** Erre kattintva egy linket fog kimásolni, amit mások megnyitva elérhetik a listádat. Csak akkor fog működik a megosztott link, ha a lista publikus és/vagy jogosultsága van a megtekintéshez az elküldött felhasználónak.



7. ábra: Lista készítés

### Karakterek

A karakterek a szinteken megjelenő elemek, alapértelmezetten csak egy képként jelennek meg.

Két funkció elérhető:

* **Információ kinyitás:** Jobb kattintással kinyitható az információs rész, ahol a megadott adatok láthatóak, valamint itt érhető el a módosítása is. Bal oldalon alul található a karakter és anime URL ikonok, amire kattintva külön oldalon fogja megnyitni a megadott linket. Bezárni az „X”-el vagy újbóli jobb kattintással tudjuk.
* **Mozgatás:** Hosszan lenyomva a bal egér gombot tudjuk áthúzni a karaktert a kiválasztott helyre, akár szinten belül, akár másik kategóriába. Amint elengedjük, már az új helyére is fog kerülni.

8. ábra: Karakter

### Karakter hozzáadása

Az üres kategóriában található „+” gombra kattintva tudjuk megnyitni az új karakter hozzáadása ablakot. Jobb felül kereshetünk karakterekre név alapján, amikből választva kitölti a mezőket. Karakter hozzáadáshoz öt adatot tudunk megadni:

* **Név:** Karakter neve, kötelező mező. Ha nem írunk semmit a mezőbe, akkor „Üres mező!” hibaüzenetet kapunk.
* **URL:** A karakterről egy weboldal URL, ahol leírás található, például MyAnimeList. Nem kötelező adat. Amennyiben adunk meg értéket, akkor hibás URL formátumnál „Nem megfelelő URL formátum!” hibaüzenetet kapunk.
* **Kép:** Karakter képe. Belinkelhető és feltölthető a kép. Kötelező mező, ha nem adjuk meg akkor „Üres mező!”, nem megfelelő belinkelt kép esetén „Az URL-en nincs elérhető kép!” hibaüzeneteket kaphatunk.

A következők a karakterhez tartozó anime adatai, amik nem kötelező mezők.

* **Cím:** Anime címe**.**
* **URL:** Az animéről egy weboldal URL-je, ahol leírás található, például MyAnimeList. Hibás link megadásakor „Nem megfelelő URL formátum!” kapunk.

9. ábra: Új karakter

### Karakter módosítás

A karakter információknál a jobb alsó sarokban tudjuk elérni a szerkesztést. Ekkor az adatok módosíthatóvá válnak. A karakter és a hozzá tartozó anime külön módosítható, ezek között jobb oldalon alul lehet váltogatni. Menteni és törölni a bal alsó sarokban található ikonokra kattintva tudunk. A karakterünk adatainak módosítása és törlése nem visszavonható. Ha mégsem szeretnénk módosítani, akkor a jobb felső sarokban található „X”-re kattintva visszatérhetünk az információk megtekintéséhez. Az itt lévő mezők megegyeznek a hozzáadásnál lévőkkel, de itt nem jelenik meg hibaüzenet. A hibás értékeknél egy piros „X” jelenik meg.



10. ábra: Karakter módosítás

### Kategóriák

A kategóriák a listákban található szintek, amikbe kerülnek a karakterek. Három fő részből áll:

* **Mozgatás:** A bal oldali két csíkos sáv segítségével áthúzható a kategória. Legalább mozgatási jogosultság kell hozzá, hogy használható legyen.
* **Cím:** Egy négyzet, aminek háttérszíne a kategória színe, rajta pedig a címe. A szöveg alkalmazkodik a háttérhez, így az mindig látható és olvasható.
* **Karakterek:** Az összes behúzott karakter itt lesz.



11. ábra: Kategória

### Kategória hozzáadása

A kategóriák alatt található „Kategória hozzáadás” gombra kattintva megjelenik a létrehozó ablak, ahol két adatot tudunk megadni:

* **Cím:** Kategória címe, ez fog megjelenni a létrejött szinten. Kötelező mező.

**Lehetséges hibaüzenetek:**

Üres mező!

* **Szín:** Kategória színe, tetszés szerint állítható. Szöveg színe alkalmazkodik a háttérhez.

Az új kategória az összes többi alatt fog megjelenni, ahonnan bármikor áthelyezhetjük. Az előnézetben látni fogjuk a beírt címet is.

12. ábra: Új kategória

### Kategória módosítás

A kategóriára ráhúzva az egerünket megjelenik a szerkesztés ikon () a kategória címénél. Erre kattintva tudjuk módosítani a szintünk címét és színét. Hibaüzenetek a karakter módosításhoz hasonlóan itt se jelennek meg, egyedüli hiba lehetőség az „Üres mező!”. Menteni a bal alsó sarokban lévő zöld mentés ikonnal (), törölni a mellette található piros szemetesre () kattintva tudunk. Az „X”-re kattintva a módosítások nem hajtódnak végre, visszaáll az eredeti állapotába.

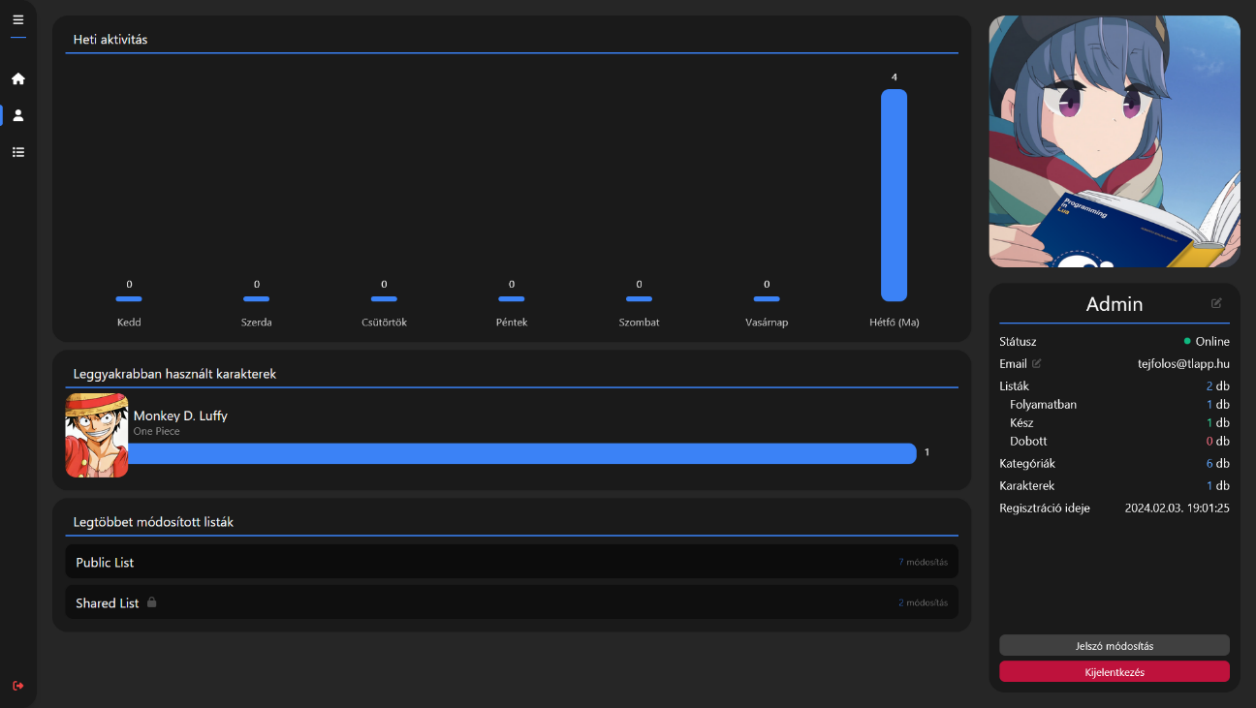
A törlés esetén a benne található összes karakter is elvész, ezért érdemes előtte áthelyezni azokat, amikre később még szükségünk lehet, mivel ez a művelet nem visszavonható.



13. ábra: Kategória módosítás

## Profil

A profilunkon megtalálhatunk mindenféle információt, statisztikát, valamint itt módosíthatjuk a felhasználónevünket, email címünket, jelszavunkat és profilképünket. Ezt az oldalt a bal oldali navigációs sávból érhetjük el. Amikor más profiljára kerülünk, akkor az e-mail címe rejtve van, a státusznál megjeleníti, hogy elérhető (Online) vagy nem (Offline). Az oldalon más felhasználónak az adatait nem módosíthatjuk, a hozzá tartozó szerkesztési opciók se jelennek meg, valamint a privát listái se érhetők el.



14. ábra: Profil

### Statisztikák

Három féle statisztikai adat jelenik meg az oldalon:

* **Heti aktivitás:** Látható rajta az elmúlt egy heti aktivitásunk. Egy aktivitásnak számít, amikor létrehozunk, mozdítunk, módosítunk, törlünk valamit a listában. Például, ha hozzáadunk 10 karaktert és 3 kategóriát, amik közül 5 karaktert áthelyezünk, akkor az 18 aktivitásnak számít arra a napra. Az egeret egy aktivitási oszlopra ráhúzva kiírja, hogy az adott napon melyik listában mennyi módosítás történt.
* **Leggyakrabban használt karakterek:** Azon karakterek listája, amelyeket a legtöbbször használtunk. Ez az összesen létrehozott karakterek száma, amit töröltünk az nem kerül bele a számításba. A képre kattintva elérünk egy képnézegetőt, amivel közelebbről is megvizsgálhatjuk a karakterünk kinézetét. A top 10 leghasználtabb fog itt megjelenni.
* **Legtöbbet módosított listák:** A listához jogosult felhasználók által elvégzett összes aktivitás alapján számolt legtöbbet módosított top 10 lista. Ezekre rákattintva – ha a saját profilunkon vagyunk vagy nem privát a lista – megnyithatjuk.

### Adatok módosítása

Az adatok módosítása felugró ablakokban történik, minden mező megadása kötelező. Amennyiben mégsem szeretnénk szerkeszteni az adatot, a „Mégsem” gombra kattintva bezárhatjuk.

* **Felhasználónév:** A felhasználónevünk mellett található szerkesztés ikonra  kattintva érjük el. Már létező felhasználók neveit nem adhatjuk meg.

15. ábra: Felhasználónév módosítás

* **Email:** Az email címünk melletti ikonnal szerkeszthető. Miután megváltoztattuk, hitelesíteni kell az új emailt, erre kapunk is egy üzenetet a megadott címre. Eredeti jelszó megadása kötelező.

16. ábra: Email módosítás

* **Jelszó:** Az új jelszónak a már regisztrációkor is meglévő feltételeknek kell megfelelnie: legalább 8 karakter hosszúság kötelező, ajánlott speciális karaktereket és számokat beleírni. Eredeti jelszó magadása kötelező.

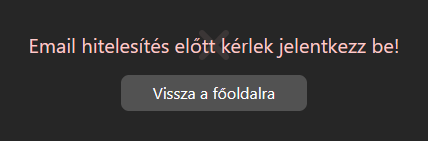
17. ábra: Jelszó módosítás

* **Profilkép:** A profilképünkre ráhúzva az egeret megjelenik a „Profilkép módosítása” felirat. Lehetőségünk van feltölteni egy képfájlt, amit ezt követően láthatunk előnézetben. A kiválasztott kép képaránya és felbontása nincs meghatározva, a négyzetet teljesen kitöltve jelenik meg. Amennyiben másik fájlt szeretnénk választani, kattintsunk újra a „Feltöltés” gombra, ezzel törölve az előző fájlunkat. A „Törlés” gomb azonnal eltávolítja a profilképünket.

18. ábra: Profilkép módosítás

## Email hitelesítés

A megadott email címre kapott üzenetben találunk egy hitelesítő linket, amit megnyitva – ha már bejelentkeztünk az oldalra – tudjuk hitelesíteni. Amennyiben nem vagy bejelentkezve, az oldal jelezni fogja számodra.



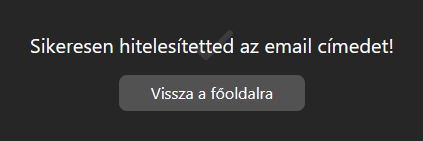
19. ábra: Email hitelesítés - Nincs bejelentkezve

Email módosítás után új hitelesítő linket kap, az előzővel már nem fog működni a hitelesítés. Megfelelő linket megnyitva, bejelentkezve sikeresen hitelesítheti az e-mail címét. Ekkor már semmi egyéb teendő nincs ezzel kapcsolatban, amíg meg nem újítja azt. Amennyiben újra próbálná hitelesíteni az e-mail címét, azt az oldal jelezni fogja.

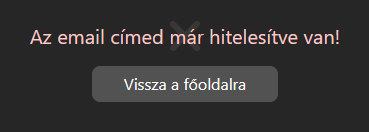
A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

20. ábra: Email hitelesítés - Hibás link



21. ábra: Email hitelesítés - Sikeres

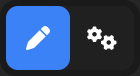


22. ábra: Email hitelesítés - Már hitelesítve van

## Navigáció

A navigációs sáv egy Sidebar, ami az oldal bal oldal található, alapértelmezetten csukott formában. Amikor ráhúzzuk az egeret, kinyitódik és láthatjuk a menüpontokat. Az öt legutóbb frissített lista is megjelenik, mellette a módosító profilképe. Alul lehetőségünk van kijelentkezni.

A listák oldalán, már megnyitott listánál megjelenik jobb oldalon alul egy két lehetőséget kínáló navigációs doboz. Itt tudunk a lista szerkesztése (bal oldali) és a lista adatok módosítása (jobb oldali) oldalak között mozogni. Ezeket a gombokat addig nem látjuk, amíg meg nem nyitottunk egy listát.



24. ábra: Lista navigáció

23. ábra: Sidebar

# Fejlesztői dokumentáció

A weboldal frontend része Vite környezetben, React pluginnal készült. A backendet egy Node JS szerver biztosítja. A projekt JavaScript nyelvben íródott, az e-mail sablonok HTML-ben készültek el. A frontend és backend egyaránt Node alapú, ezért mindkettő Node csomagokat használ.

## Telepítés

Az oldal mindkét része Node alapú, ezért hasonló indítási lehetőségekkel rendelkeznek. A projekt több csomagkezelővel is telepíthető, az általam használtak és teszteltek az Npm és a Bun. Végleges verzióban én a Bun-t használom. Futtatás előtt ajánlott a dotenv fájlt bekonfigurálni, mivel anélkül problémák keletkezhetnek, ennek leírása megtalálható a Backend, Környezeti változók cím alatt.

### Npm

Az Npm a Node JS-hez tartozó csomagkezelő. Minden hozzá tartozó parancs az npm kulcsszóval kezdődik.

25. ábra: NPM

**Weboldal:** <https://nodejs.org/en>, <https://www.npmjs.com/>

### Bun



A Bun egy különálló csomagkezelő. A Bun jóval gyorsabban telepíti a csomagokat, mint az npm, ezért sok időt megspórolhatunk. Parancsai a bun kulcsszóval kezdődnek.

26. ábra: Bun

**Weboldal:** <https://bun.sh/>

### Telepítési parancsok

|  |  |
| --- | --- |
| **Npm** | **Bun** |
| **Frontend** | |
| npm install  npm run build  npm run dev | bun install  bun run build  bun run dev |
| **Backend** | |
| npm install  npm run start  npm run dev | bun install  bun run start  bun run dev |

## Fejlesztés során használt szoftverek

Fejlesztés során sok szoftvert használhatunk, ami könnyíti a folyamatot. Tervezéshez, kódíráshoz, feladatok követésére és tesztelésre is találhatunk szoftvereket, weboldalakat.

### Visual Studio Code (^1.88.0)

A Visual Studio Code egy ingyenes, nyílt forráskódú kódszerkesztő. A kódot jobban átlátjuk benne, mint egy normál szövegszerkesztőben és a hibakezelési folyamatot is elősegíti. A beépített terminálon keresztül indíthatjuk a szükséges szervereket, telepíthetünk csomagokat, használhatunk hasznos bővítményeket.

Az alábbi bővítményeket használtam a fejlesztés során:

* **ES7 React/Redux/GraphQL/React-Native snippets:** Frontendhez, a React pluginnál lett felhasználva, egyszerűbbé teszi a fejlesztést.
* **Console Ninja:** Amikor futtatjuk a programunkat, előfordulhat, hogy consoleban kiírunk változókat tesztelés miatt, ezeket láthatjuk a kódunkban is, ott ahol alkalmaztuk a logolást.
* **Tailwind CSS IntelliSense:** A design elkészítésére a Tailwindet használtam, ezt leegyszerűsítve töltöttem le ezt a bővítményt, ami kiegészíti a class neveket.
* **Thunder Client:** Backend útvonalak tesztelésére használt bővítmény.

### XAMPP (^3.3.0)

Az XAMPP egy platformfüggetlen webszerver-szoftvercsomag. Ebben a projektben a MySQL adatbázis szervert biztosítja, valamint az Apache szerver elindításával a phpMyAdmin oldalon láthatóak és kezelhetőek az adatok.

### Git & Github

A Git egy verziókezelő szoftver, ami lehetőséget biztosít a projekt módosításainak kezelésére és figyelésére. A Github egy adattároló weboldal, ahol láthatjuk a Git segítségével feltöltött adatainkat. A projektünk változtatásainak megfigyelésére és távoli elérésére is alkalmas.

### Atlassian Jira

A Jira lehetővé teszi a projektünk menedzselését, a hibák és problémák követését. Minden fontosabb feladat megtalálható a projekt Jira oldalán. Láthatjuk a teendő, folyamatban lévő és kész feladatainkat, így könnyen szűrhetjük, hogy mit kell még megcsinálni. Bármikor bővíthető, így ha új funkciók, elvárások kerülnek a projektbe, akkor probléma nélkül hozzáadhatjuk.

### Figma

A Figma egy design tervező szolgáltatás. A weboldal összes alapterve ebben a szoftverben készült, ezek megtalálhatóak a projekt Github oldalán is.

### Firefox (^124.0.2)

A Firefox egy böngésző, amit a weboldal tesztelésére használtam. A fejlesztői konzol lehetőséget biztosít a funkciók, konzolra kiírt üzenetek, hibák és tárolt adatok figyelésére is.

## Frontend

A frontend Vite környezetben készült React plugin felhasználásával. A Vite egy gyors fejlesztői környezet, amely sokféle projektet támogat. A React egy JavaScript könyvtár, ami lehetővé teszi, hogy az oldalon megjelenő tartalom újra töltés nélkül látható legyen.

### Függőségek

A felsorolt csomagok elérhetőek az npmjs.com oldalon.

Node verzió: v20.10.0  
Npm verzió: v10.2.3  
Bun verzió: v1.1.3

* **@dnd-kit/core (^6.1.0):** Drag and drop React könyvtár
* **@dnd-kit/modifiers (^7.0.0):** Dnd-kit mozgatást módosító eszközök
* **@dnd-kit/sortable (^8.0.0):** Sorbarendezős dnd-kit interfészek
* **@dnd-kit/utilities (^3.2.2):** Segédeszközök a dnd-kithez
* **@fortawesome/free-solid-svg-icons (^6.5.2):** FontAwesome ikon csomag
* **@fortawesome/react-fontawesome (^0.2.0):** React FontAwesomeIcon, az ikonok használatához
* **axios (^1.6.8):** Backend-frontend kapcsolat létrehozása
* **moment (^2.30.1):** Idő formátumok és műveletek
* **react (^18.2.0):** React könyvtár a keretrendszerhez
* **react-device-detect (^2.2.3):** Lekérhető, hogy a felhasználó milyen eszközről használja a weboldalt (telefon, számítógép)
* **react-dom (^18.2.0):** Reacthoz szükséges függőség
* **react-router-dom (^6.22.3):** React útvonalak kezelése
* **react-tooltip (^5.26.3):** Információ, eszköztipp jelenik meg egy-egy elemre ráhúzva az egeret
* **socket.io-client (^4.7.5):** Kétirányú eseményalapú kommunikációk kezelése kliens oldalon
* **use-debounce (^10.0.0):** Gépelés befejezésének megvárására használt csomag
* **postcss (^8.4.38):** Tailwindhez szükséges, JavaScript segítségével lehet módosítani a CSS-t.
* **tailwindcss (^3.4.3):** Az oldal felületének designjáért felelős CSS keretrendszer.
* **vite (^5.2.0):** Gyors fejlesztői környezet.

### Vite



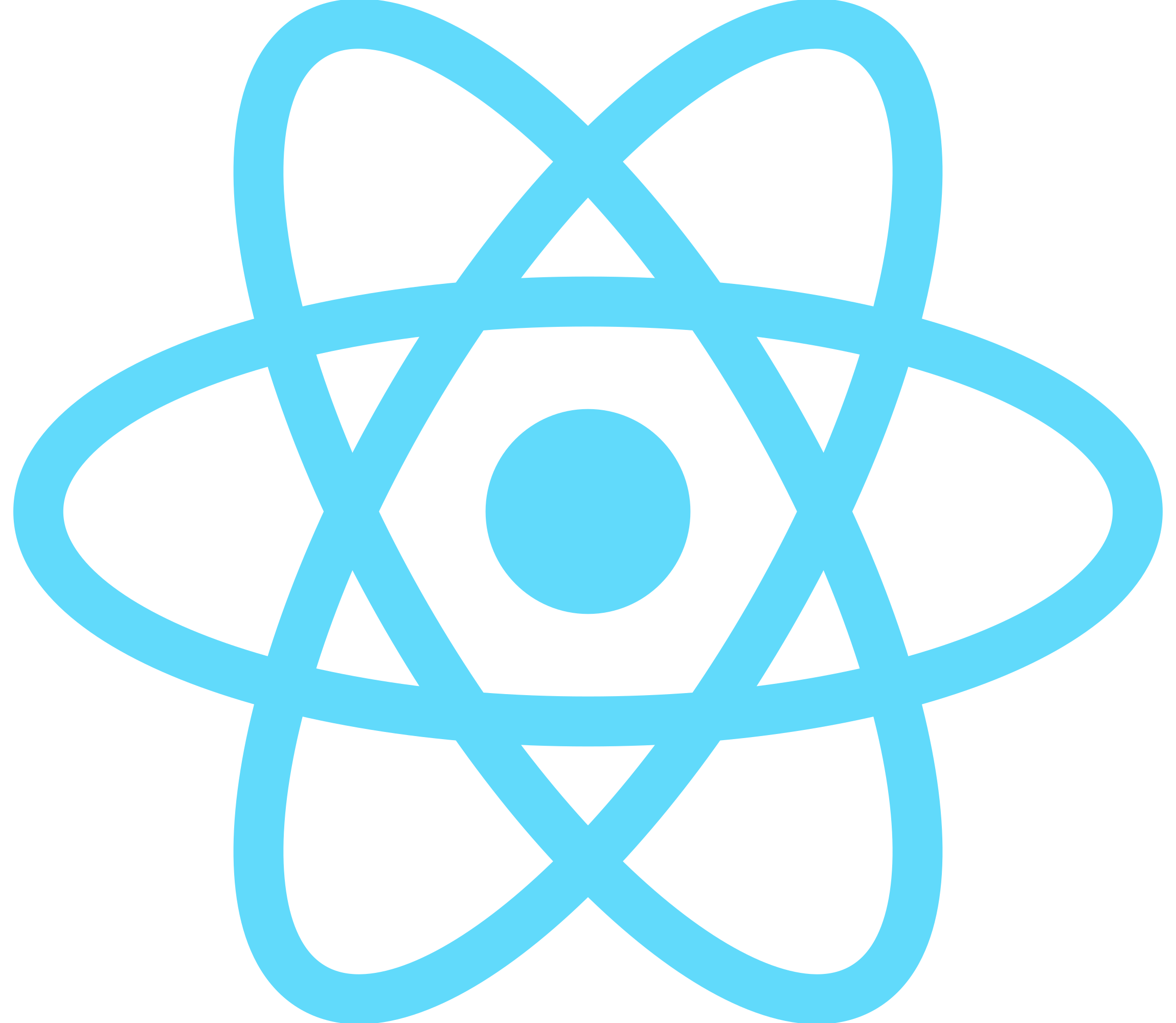
A Vite egy gyors és modern webfejlesztési környezetet biztosít a fejlesztők számára. Lehetőséget nyújt sokféle keretrendszer és könyvtár használatához, például Vue, Preact, és a projektem által használt React is. A Vite használható npm és bun csomagkezelővel egyaránt.

27. ábra: Vite

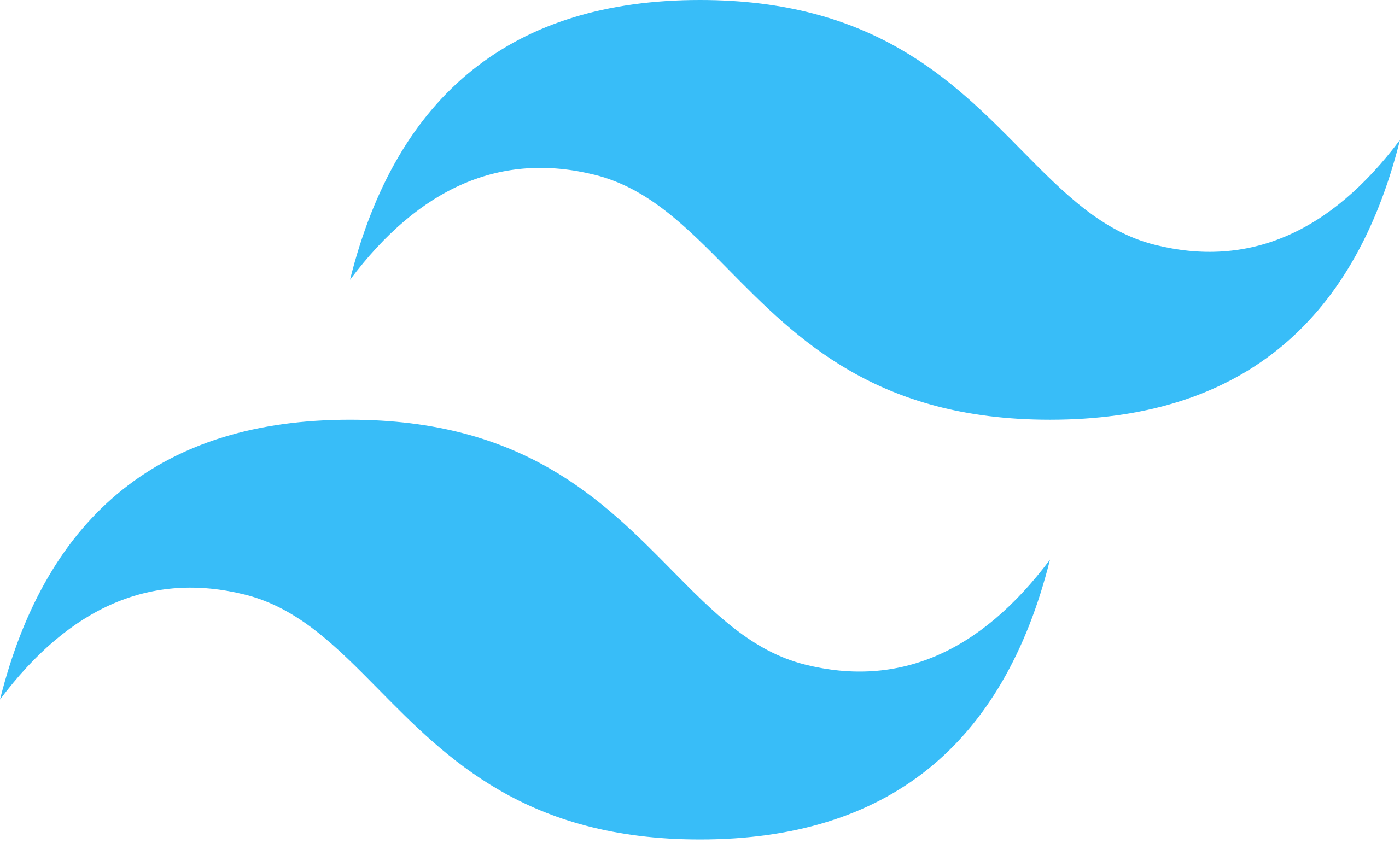
### React

A React egy komponens alapú JavaScript frontend könyvtár. Webes és natív interfészeket is létrehozhatunk vele. Interaktívvá tehetőek vele az oldalunk található elemek.

28. ábra: React



### Tailwind



A Tailwind egy CSS keretrendszer, ami lehetővé teszi, hogy egyszerűen designolhassuk az oldalunkat. Class neveket megadva alakíthatjuk a teljes kinézetét az oldalnak.

29. ábra: Tailwind

### dnd kit



Reacthoz készült könnyű, moduláris, nagy teljesítményű, hozzáférhető, bővíthető és könnyen szerkeszthető drag and drop eszközkészlet.

30. ábra: dnd kit

### FontAwesome

A FontAwesome egy internetes ikon könyvtár. Mindenféle kategóriában találhatunk itt ikonokat, bármilyen oldalra beépíthetőek és elérhető sokféle keretrendszerben is.

31. ábra: FontAwesome

## Backend

A backend egy Node JS szerveren fut. A legtöbb útvonal védve van a nem bejelentkezett felhasználók ellen, a főoldal az egyetlen regisztrációt nem igénylő oldal. A többfelhasználós használathoz tartozó socket szerver is itt fut.

### Függőségek

A felsorolt csomagok elérhetőek az npmjs.com oldalon.

Node verzió: v20.10.0  
Npm verzió: v10.2.3  
Bun verzió: v1.1.3

* **bcrypt (^5.1.1):** Jelszavak titkosítása.
* **chalk (^5.3.0):** Console log színezés, jobban értelmezhető konzol elérése.
* **cookie-parser (^1.4.6):** Sütik kezelése.
* **cors (^2.8.5):** Cross Origin kérések engedélyezése.
* **dotenv (^16.3.1):** Környezeti változók kezelése.
* **express (^4.18.2):** HTTP szerver.
* **jsonwebtoken (^9.0.2):** Session és bejelentkezés kezelés.
* **moment (^2.30.1):** Idő formátumok és műveletek.
* **multer (^1.4.5):** Fájlfeltöltések kezelése.
* **mysql2 (^3.6.5):** Adatbázis-kezelés, sequelizehoz kapcsolódó csomag.
* **resend (^3.2.0):** E-mail küldések kezelése.
* **rndstr (^1.0.0):** Random string generátor.
* **sequelize (^6.35.2):** Adatbázis kezelése.
* **socket.io (^4.7.4):** Kétirányú eseményalapú kommunikácók kezelése.
* **nodemon (^3.0.2):** Fejlesztői futtatáshoz tartozó fejlesztői függőség.

### Környezeti változók (.env)

A backendhez tartozó környezeti változókat kötelező definiálni, mivel enélkül a program nem futtatható, hibákba fog ütközni. A működő adatokat tartalmazó dotenv megtalálható a backend mappájában **.env** vagy **.env.example** néven. Amennyiben a Github törölné a .env fájlt, akkor a .env.example-t csak át kell nevezni .env-re.

* **PORT:** A backend ezen a porton lesz elérhető futtatás után. Ajánlott az alapértelmezett portot használni, mivel a frontend is azt fogja keresni. Alapértelmezett: 2000
* **DB\_HOST:** Adatbázist elérő URL. Alapértelmezetten: localhost
* **DB\_USER:** Adatbázist elérő felhasználó neve. Alapértelmezetten: root
* **DB\_PASS:** Adatbázist elérő felhasználó jelszava. Alapértelmezetten üres, mivel localhoston a root felhasználóhoz nincs jelszó.
* **DB\_NAME:** Adatbázis neve. Alapértelmezetten: tier-list
* **ACCESS\_SECRET:** Egy véletlenszerűen generált token. Üres nem lehet.
* **RESEND\_KEY:** Resend mailer kulcs. Ehhez a projekthez van egy fenttartott kulcsom: re\_4DRKtUsP\_Hma4jULpQXoCay5kwRjy8NMm

### Adatbázis

Az adatbázishoz az XAMPP v3.0.0 és a hozzá tartozó phpMyAdmin oldalát használtam, a MySQL verzió 8.2.12. Az egész adatbázist, táblákat és kapcsolatokat a Sequelize adatbázis kezelő csomag létrehozza, ezért azzal külön foglalkozni nem szükséges.

**ER modell:**

A képen kör, képernyőkép, fekete-fehér, Grafika látható

Automatikusan generált leírás

32. ábra: ER Modell

**Adatbázis tábla kapcsolatok és relációs séma:**

A képen szöveg, képernyőkép, Téglalap, fekete látható

Automatikusan generált leírás

33. ábra: Relációs séma

A táblázatban a vastag betűs sorok az elsődleges kulcsokat, a dőltek az idegen kulcsokat jelölik. Oszlopok tartalma sorban: mező neve, típusa (zárójelben a hossza) és egy rövid leírás.

* **users:** Felhasználó adatai

|  |  |  |
| --- | --- | --- |
| **id** | **Szám** | **Felhasználó azonosító** |
| username | Szöveg (128) | Felhasználónév |
| email | Szöveg (128) | Email cím |
| password | Szöveg (128) | Titkosított jelszó |
| avatar | Szöveg (256) | Profilkép útvonal |
| status | Szám | Online / Offline állapot |
| accessToken | Szöveg (256) | Bejelentkezési token |
| verifyToken | Szöveg (64) | Email hitelesítő token |
| registerDate | Dátum | Regisztráció dátuma |

* **lists:** Felhasználó listái

|  |  |  |
| --- | --- | --- |
| **id** | **Szám** | **Lista azonosító** |
| name | Szöveg (128) | Lista címe / neve |
| description | Szöveg (256) | Leírás |
| status | Szám (1) | Státusz |
| private | Logikai | Privát-e a lista |
| *userId* | *Szám* | *Felhasználó azonosító* |

* **permissions:** Felhasználók jogosultságai a listákhoz

|  |  |  |
| --- | --- | --- |
| value | Szám (1) | Jogosultság fajtája |
| *userId* | *Szám* | *Felhasználó azonosító* |
| *listId* | *Szám* | *Lista azonosító* |

* **updates:** Lista aktivitások

|  |  |  |
| --- | --- | --- |
| **id** | **Szám** | **Azonosító** |
| count | Szám | Módosítás darabszám |
| date | Dátum | Módosítás dátuma |
| time | Idő | Módosítás ideje |
| *userId* | *Szám* | *Felhasználó azonosító* |
| *listId* | *Szám* | *Lista azonosító* |

* **categories:** Lista kategóriák

|  |  |  |
| --- | --- | --- |
| **id** | **Szám** | **Kategória azonosító** |
| name | Szöveg (32) | Kategória neve |
| position | Szám | Listán belüli pozíció |
| color | Szöveg (32) | Színkód |
| *listId* | *Szám* | *Lista azonosító* |

* **characters:** Kategórián belüli karakterek

|  |  |  |
| --- | --- | --- |
| **id** | **Szám** | **Karakter azonosító** |
| name | Szöveg (256) | Karakter neve |
| position | Szám | Kategórián belüli pozíció |
| url | Szöveg (512) | Karakter URL |
| image | Szöveg (512) | Kép URL |
| *categoryId* | *Szám* | *Kategória azonosító* |
| *animeId* | *Szám* | *Anime azonosító* |

* **animes:** Karakterhez tartozó anime

|  |  |  |
| --- | --- | --- |
| **id** | **Szám** | **Anime azonosító** |
| title | Szöveg (256) | Anime címe |
| url | Szöveg (512) | Anime URL |

### Útvonalak

* GET / – Gyökér útvonal

**Felhasználói útvonalak**

* GET /logged – Bejelentkezési állapot lekérés
* POST /register – Regisztráció
* POST /login – Bejelentkezés
* DELETE /logout – Kijelentkezés
* GET /user/data – Felhasználói adatok lekérése
* GET /user/token/refresh – Token frissítés
* GET /user/lists – Felhasználó listáinak lekérése
* GET /user/lists/:id – Lista, kategóriái és karaktereinek lekérdezése
* PATCH /user/username – Felhasználónév módosítás
* PATCH /user/avatar – Profilkép módosítás
* DELETE /user/avatar – Profilkép törlése
* PATCH /user/email – Email módosítás
* PATCH /user/password – Password módosítás
* POST /user/email/verify – Email hitelesítés

**Lista útvonalak**

* GET /lists/sidebar – Navigációs menüben megjelenő listák
* GET /lists/shared – Megosztott listák
* GET /lists/public – Véletlenszerűen kiválasztott 10 publikus lista
* POST /lists/create – Lista létrehozás
* PATCH /lists/:id/update – Lista módosítás
* DELETE /lists/:id/remove – Lista törlés

**Lista jogosultság útvonalak**

* POST /lists/:id/permissions/create – Jogosultság létrehozás
* PATCH /lists/:id/permissions/update/:userId – Jogosultság módosítás
* DELETE /lists/:id/permissions/remove/:userId – Jogosultság törlés

**Lista kategória útvonalak**

* POST /lists/:id/categories/create – Kategória létrehozás
* PATCH /lists/:id/categories/:categoryId/move – Kategória mozgatás
* PATCH /lists:/id/categories/:categoryId/update – Kategória módosítás
* DELETE /lists/:id/categories/:categoryId/remove – Kategória törlés

**Lista karakter útvonalak**

* POST /lists/:id/characters/create – Karakter létrehozás
* PATCH /lists/:id/characters/:characterId/move – Karakter mozgatás
* PATCH /lists/:id/characters/:characterId/update – Karakter módosítás
* DELETE /lists/:id/characters/:characterId/remove – Karakter törlés

**Fájl lekérés útvonalak**

* GET /user/images/:filename – Profilkép lekérése
* GET /characters/images/:filename – Karakter kép lekérése

## Fájlrendszer

A projekt három fő mappából áll: backend, frontend és documents. A documents mappában található az összes dokumentáció, tervezések és minden hozzá tartozó kép és egyéb fájl. A benne található gitignore azokat a fájlokat tartalmazza, amik csak ideiglenesen készültek el, valahol felhasználásra kerültek és nincs bennük fontos adat.

### Backend mappák:

* **controllers:** Minden, amit valahogy kezelni kell, annak a függvénye itt található valamely fájlban. A controller.js és socketController.js tartalmazza az összes útvonalakhoz és socket szerverhez tartozó funkciókat. Két fájlban csak middleware található, a checkPermission.js-ben a jogosultságokat tesztelem, míg a verifyToken.js a felhasználói bejelentkezéseket ellenőrzi.
* **images:** Profilképek és karakter képek. Amit a felhasználók feltöltenek, azok ide kerülnek.
* **libs:** Saját könyvtárak, amik elősegítik az egyszerűbb kódolást.
* **models:** Adatbázis tábla modellek. Az összes táblának megtalálható a sequelize-hoz szükséges modellje.
* **node\_modules:** Ez a mappa csak a függőségek telepítése után fog megjelenni, tartalmazza az összes node csomagot. (npm és bun parancsos telepítéssel is meg fog jelenni)
* **routes:** Backend útvonalakat tartalmazza.
* **sql:** Előre elkészített adatbázisok, nem szükséges beimportálni, kivéve, ha nem bízunk a sequelize adatbázis létrehozásában vagy előre beállított adatokra van szükségünk.
* **Egyéb fájlok:** Itt található a .env, .env.example a környezeti változókhoz. Az index.js és socket.js fájlok tartalmazzák a szerverek alapbeállításait.

### Frontend mappák:

* **node\_moludes:** Hasonlóan a backendhez, csak akkor fog megjelenni, ha letöltöttük a függőségeket.
* **public:** Publikus fájlok, ikonok.
* **src:** A weboldal összes aloldala, komponense és stílusai.
* **components:** Apróbb komponensek az oldalhoz.
* **css:** App.css, tailwind és animációk stílusai.
* **pages:** Az oldalak fő fájljai.
* **Egyéb fájlok:** Index.html, PostCSS, TailwindCSS és Vite konfigurációs fájlok.

## Továbbfejlesztési lehetőségek

A projekt elkészítése közben az is egy cél volt, hogy a későbbiekben egyszerűen továbbfejleszthető legyen. Az oldalak komponensekre vannak bontva, így fájl megnyitás nélkül megtalálhatunk egyes részeket. A kód a fontosabb részeken leírásokat – kommenteket – tartalmaz, hogy keresés közben tudjuk tájékozódni.

### Főoldal Demo

A főoldalon jelenleg nem sok információ található az oldalról, valamint kevés a motivációs tényező a regisztrációra. Lejjebb görgetés esetén egy kevesebb funkciós tier list teszt verzió feldobhatná az oldalt, kedvet adhatna a felhasználóknak a regisztrációra.

### Beállítások

Legfontosabb beállítások, mint az e-mail megjelenés, profilon megjelenő adatok, privát profilra váltás, felhasználók letiltása és bejelentkezett eszközök kezelése kerülne ide elsősorban. Ez a lehetőséget adna az oldalt használóknak, hogy testre szabhassák a profilukon megjelenő tartalmakat.

### Téma mentes Tier List

Az oldal jelenleg kizárólag anime témában ad lehetőséget tier list elkészítésére. Szélesebb körű felhasználóbázis elérése érdekében lehetne több témát és egy téma mentes lista készítési opciót is biztosítani a felhasználók számára. Ezáltal mindenki azt készíthet, amit tényleg szeretne.

### Barátlista, csevegő

Az oldalon jelenleg nincs lehetőség a felhasználóknak egymással kommunikálni. Lista készítés közben egy helyi csevegő jelenhetne meg a jobb alsó navigációs menü opciói között, amit ki lehetne nyitni. Amennyiben lehetnek barátaik az oldalon, akkor egy privát csevegőt is hozzá lehetne adni. Egyszerűbbé téve a barátaink megtalálását, egy külön felhasználó kereső oldal is megvalósítható.

### Elfelejtettem a jelszavam

A bejelentkezésnél egy „Elfelejtettem a jelszavam” opció hasznos lenne, hogy aki tényleg ilyen problémával küzd, az meg tudja változtatni a jelszavát és ezáltal elérhesse a profilját. E-mail címére kapott jelszó változtató linkre kattintva, eljutna a weboldalra, ahol megadhatná az új jelszavát.

# Összegzés

# Ábrajegyzék

1. ábra: Főoldal 8

2. ábra: Regisztráció 9

3. ábra: Bejelentkezés 10

4. ábra: Listák 11

5. ábra: Új lista 11

6. ábra: Jogosultságok 12

7. ábra: Lista készítés 14

8. ábra: Karakter 14

9. ábra: Új karakter 16

10. ábra: Karakter módosítás 16

11. ábra: Kategória 17

12. ábra: Új kategória 17

13. ábra: Kategória módosítás 18

14. ábra: Profil 18

15. ábra: Felhasználónév módosítás 19

16. ábra: Email módosítás 20

17. ábra: Jelszó módosítás 20

18. ábra: Profilkép módosítás 21

19. ábra: Email hitelesítés - Nincs bejelentkezve 21

20. ábra: Email hitelesítés - Hibás link 22

21. ábra: Email hitelesítés - Sikeres 22

22. ábra: Email hitelesítés - Már hitelesítve van 22

23. ábra: Sidebar 23

24. ábra: Lista navigáció 23

25. ábra: NPM 23

26. ábra: Bun 24

27. ábra: Vite 26

28. ábra: React 27

29. ábra: Tailwind 27

30. ábra: dnd kit 27

31. ábra: FontAwesome 27

32. ábra: ER Modell 29

33. ábra: Relációs séma 29