



ZÁRÓDOLGOZAT

Készítették:

Bodnár István Gábor - Fekete László - Takács Krisztián

Konzulens:

Németh Bence

Miskolc

2024

Miskolci SZC Kandó Kálmán Informatikai Technikum

Miskolci Szakképzési Centrum

SZOFTVERFEJLESZTŐ- ÉS TESZTELŐ SZAK

Trillence

Zene streaming szolgáltatás

Bodnár István Gábor - Fekete László - Takács Krisztián

2023-2024

Tartalomjegyzék

1. Feladatkörök	5
1.1. Bodnár István Gábor	5
1.2. Fekete László	5
1.3. Takács Krisztián	5
2. Fejlesztői környezet bemutatása	6
2.1. Visual Studio 2022	6
2.2. Visual Studio Code	6
2.3. MySQL - phpmyadmin	6
3. Használt technológiák	7
3.1. ReactJS	7
3.2. ASP.NET WebAPI	7
3.3. Github	7
3.4. Bootstrap CSS - FontAwesome Icons	7
3.5. MySql	7
3.6. Npm	8
4. Deploy platform	9
4.1. Linux otthoni szerver	9
4.2. Cloudflare - Zero trust tunnel és Domain	9
5. A Program	10
5.1. A Program leírása	10
5.2. A Program működése	10
6. Tájékozódás az oldalon belül	11
6.1. Főoldal	11
6.2. Zenelejátszó	11
6.3. Navigációs menü	13
6.3.1. Kezdőlap	13
6.3.2. Kereső	13
6.3.3. Lejátszási lista	13
6.4. Saját Profil	13
6.4.1. Jelszó módosítása	13
6.4.2. Profilnév módosítása	13
6.5. Admin panel	14
6.6. SpotDL	14
6.6.1. SpotDL Használata	14
6.7. Bejelentkezés	14
6.7.1. Elfelejtett jelszó	14
6.8. Regisztráció	15
6.8.1. Regisztráció menete és követelményei	15

6.8.2. E-mail megerősítés	15
6.9. Kijelentkezés	15
7. Backend	16
7.1. authorization végpont:	16
7.1.1. register	16
7.1.2. AssignRole	16
7.1.3. Login	16
7.1.4. Change Username	16
7.1.5. Change Password	16
7.1.6. send-reset-email	16
7.1.7. reset-password	16
7.1.8. delete-user	16
7.1.9. is-admin	16
7.2. Backend végpont:	18
7.2.1. General CRUD végpontok	18
7.2.2. Albumimage	19
7.2.3. Connection	19
7.2.3.1. allsongdetails	19
7.2.3.2. songdetailsbyid/Id	19
7.2.3.3. allplaylistdetails	19
7.2.3.4. allplaylistdetailsby/Id	19
7.2.4. Metadata	19
7.2.5. Musicstreaming	20
7.2.5.1. current	20
7.2.5.2. current/details	20
7.2.5.3. next	20
7.2.5.4. previous	20
7.2.5.5. playlist/playlistid/current	20
7.2.5.6. currentplaylist/playlistid/details	20
7.2.5.7. playlist/playlistid/next	20
7.2.5.8. playlist/playlistid/previous	20
8. Tesztelés	21
8.1.1. nUnit tesztek	21
8.1.2. ReactJS JEST tesztek	21
8.1.3. Lighthouse teljesítmény eredmények	21
9. Forrásjegyzék	22

1. Feladatkörök

1.1. Bodnár István Gábor

Dokumentáció, FrontEnd wireframe, grid rendszer, Frontend logika első verziói

1.2. Fekete László

Ui megvalósítása, branding létrehozása, grafikák, frontend végső verziói

1.3. Takács Krisztián

Backend, adatbázis, frontendben a zenelejátszó

2. Fejlesztői környezet bemutatása

2.1. Visual Studio 2022

A Visual Studio egy hatékony fejlesztői eszköz, amellyel a teljes fejlesztési ciklust egy helyen végezheti el. A Visual Studio egy átfogó integrált fejlesztőkörnyezet (IDE), amellyel kódot írhat, szerkeszthet, hibakeresést végezhet és építhet, majd telepítheti az alkalmazást. A kódszerkesztésen és hibakeresésen túl a Visual Studio tartalmaz fordítókat, kódkiegészítő eszközöket, forráskód-ellenőrzést, bővítményeket és még számos olyan funkciót, amelyek a szoftverfejlesztési folyamat minden szakaszát javítják.

2.2. Visual Studio Code

A Visual Studio Code az egyik legelterjedtebb kódszerkesztő amely egy ingyenes és nyílt forráskódú, amely használható Windows, Linux és macOS operációs rendszereken. A program számos funkcionalitást kínál, beleértve a hibakeresést, szintaxis kiemelést, intelligens kódkiegészítést és beépített Git támogatást. A felhasználók testreszabhatják a kinézetet, gyorsbillentyű-kiosztást és telepíthetnek kiegészítőket további funkciókhoz és testreszabási lehetőségekhez.

2.3. MySQL - phpmyadmin

A XAMPP egy összeállított, platformfüggetlen webszerver-szoftvercsomag. Ez egy integrált rendszert alkot, mely lehetővé teszi webes alkalmazások készítését, tesztelését és futtatását egy csomagban. A XAMPP részeként elérhető a phpMyAdmin is, mely egy ingyenes, nyílt forráskódú adminisztrációs eszköz a MySQL adatbázisok kezelésére. Ez a szoftver könnyen telepíthető és használható, és kiválóan alkalmas az adatbázisok intuitív adminisztrációjára. A phpMyAdmin számos fontos funkciót kínál a felhasználóknak. Képes adatbázisok létrehozására, törlésére és módosítására, valamint táblák kezelésére, beleértve azok létrehozását, törlését és módosítását is. Emellett lehetőséget nyújt az egyes mezők módosítására, hozzáadására vagy törlésére, valamint SQL parancsok futtatására az adatbázisban.

3. Használt technológiák

3.1. ReactJS

A React egy ingyenes és nyílt forráskódú front-end JavaScript könyvtár, melynek fő célja komponenseken alapuló felhasználói felületek létrehozása, egyszerűen és gyorsan. Az egyik fő előnye, hogy csak azokat a DOM-elemeket rendezi újra, amelyek megváltoztak, ezáltal elkerülve a változatlan elemek felesleges újratöltését. A React deklaratív nyelvezete lehetővé teszi a fejlesztők számára, hogy kifejezzék a kívánt végeredményt anélkül, hogy részletesen leírják a lépéseket, ami jelentős könnyebbséget jelent a weblap fejlesztés során. A React hatékony eszköz a dinamikus weblapok fejlesztéséhez, minimalizálva a felesleges munkát és támogatva a gyors és hatékony fejlesztési folyamatot.

3.2. ASP.NET WebAPI

Az ASP.NET egy szerveroldali webalkalmazás-keretrendszer, amelyet dinamikus weboldalak létrehozására terveztek. A Microsoft fejlesztette ki annak érdekében, hogy lehetővé tegye a programozók számára dinamikus webhelyek, alkalmazások és szolgáltatások építését. Az API-k lehetővé teszik az alkalmazások közötti kommunikációt és az adatok megosztását, így segítve a fejlesztőket abban, hogy hatékony és skálázható alkalmazásokat hozzanak létre.

3.3. Github

A GitHub egy olyan platform, amely lehetővé teszi a fejlesztők számára, hogy hatékonyan együttműködjenek különböző szoftverfejlesztési projekteken. Alapja a Git elosztott verziókezelő rendszer, amely lehetővé teszi a fejlesztők számára, hogy nyomon kövessék a kódbázis változásait, és könnyen együtt dolgozzanak rajta. A GitHubon keresztül a fejlesztők tárolhatják, kezelhetik és megoszthatják kódjaikat, valamint nyomon követhetik a változtatásokat.

3.4. Bootstrap CSS - FontAwesome Icons

A Bootstrap egy ingyenes és nyílt forráskódú CSS keretrendszer, melyet reszponzív frontend webfejlesztéshez terveztek. A projekt HTML, CSS és opcionálisan JavaScript alapú tervezési sablonokat kínál tipográfiához, űrlapokhoz, gombokhoz, navigációhoz és más felületi komponensekhez. A bootstrap az egyik legelterjedtebb frontend keretrendszer

3.5. MySql

A MySQL egy népszerű, nyílt forráskódú relációs adatbázis-kezelő rendszer, melyet gyakran használnak dinamikus webhelyek szolgáltatására. Az SQL egy programozók által használt nyelv, amely lehetővé teszi relációs adatbázisok létrehozását, módosítását, adatok lekérdezését és a felhasználói hozzáférés szabályozását. A relációs adatbázisok az adatokat egy vagy több táblába szervezik, ahol az adatok kapcsolatban lehetnek egymással, ami segíti az adatok strukturálását, és kezelését.

3.6. Npm

Az npm egy olyan csomagkezelő rendszer a JavaScript programozási nyelvhez. A kliens segítségével lehetőség van a csomagok keresésére, telepítésére és frissítésére, a csomagok könnyen böngészhetők és kereshetők az npm weboldalán keresztül, így a fejlesztők gyorsan és hatékonyan megtalálhatják a számukra megfelelő modulokat és eszközöket.

4. Deploy platform

4.1. Linux otthoni szerver

Ha esetleg a helyi számítógépen való futtatással gondok lennének akkor a programnak egy működő verziója elérhető a következő weboldalon:

Trillence weboldal (katt link)

4.2. Cloudflare - Zero trust tunnel és Domain

A Cloudflare zero trust Tunnel egy biztonságos megoldást kínált nekünk. Az erőforrásaink elérhetők anélkül, hogy nyilvánosan routolható IP-címeket kellene használni. Az úgynevezett "Tunnel" segítségével a forgalom nem küldődik ki külső IP-címre, hanem egy könnyűsúlyú a "cloudflared" segítségével kimenő kapcsolatokat hoz létre a Cloudflare globális hálózatával. Ennek eredményeként HTTP és SSH kiszolgálók, távoli asztali számítógépek és más protokollok biztonságosan csatlakozhatnak a Cloudflare-hez. Ez lehetővé teszi az eredeti szerver számára, hogy kiszolgálja a Cloudflare-en keresztüli forgalmat, miközben védve van a Cloudflare-t megkerülő támadásokkal szemben.

5. A Program

5.1. A Program leírása

A Trillance egy új zenestreaming platform, melyet úgy terveztünk, hogy diszkrét megoldást kínáljunk az ingyenes zenehallgatásra, minimális helyigény mellett.

Célunk, hogy bárhol és bármikor élvezhessék a felhasználók a kedvenc dalokat, anélkül, hogy az eszközeik túlzott terhelést jelentenének.

A szolgáltatásunk szerves részét képezi egy folyamatosan bővülő zenei könyvtár, melyben mindig elérhetőek az előadók legfrissebb művei. Ennek érdekében elköteleztük magunkat abban, hogy a legújabb zenei trendekkel és kiadványokkal mindig lépést tartva frissítsük és gazdagítsuk repertoárunkat.

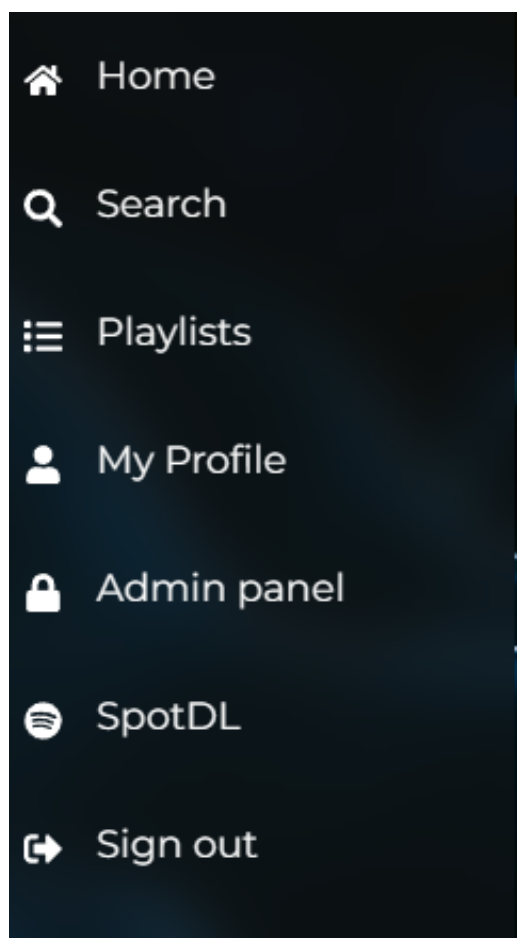
Ennek a platformnak a segítségével szeretnénk lehetővé tenni, hogy a felhasználók könnyen és kényelmesen böngésszék és hallgassák meg a zenei tartalmakat. Az intuitív felületnek és a felhasználóbarát dizájnnak köszönhetően minden eszközön élvezhető lesz a Trillance szolgáltatása.

5.2. A Program működése

6. Tájékozódás az oldalon belül

6.1. Főoldal

Ez az oldalt a nem bejelentkezett felhasználók is megtekinthetik és lejátszathatnak zenéket is. A főoldalon a felhasználó megtalálhatja az összes megtalálható zenéket. Ha lentebb görget a felhasználó és elér a látható lista végére akkor a "load more songs" gombbal tud még többet betölteni illetve megjeleníteni. a weboldal alsó sávjában megtalálható a zenelejátszó, baloldalt pedig a navigációs menü. Telefonról nézve pedig a fenti sávban található hamburger menüvel érheti el a navigációs menüt



6.2. Zenelejátszó

A zenelejátszó a weboldal alján található. Ha a felhasználó számítógépen használja az weboldalt akkor az alsó sáv bal oldalán Az album borító, a zenecíme és az előadó neve található.

Középen találhatók a zenelejátszó irányító gombjai, amelyek a következők balról jobbra haladva:

Zene Visszaugrása

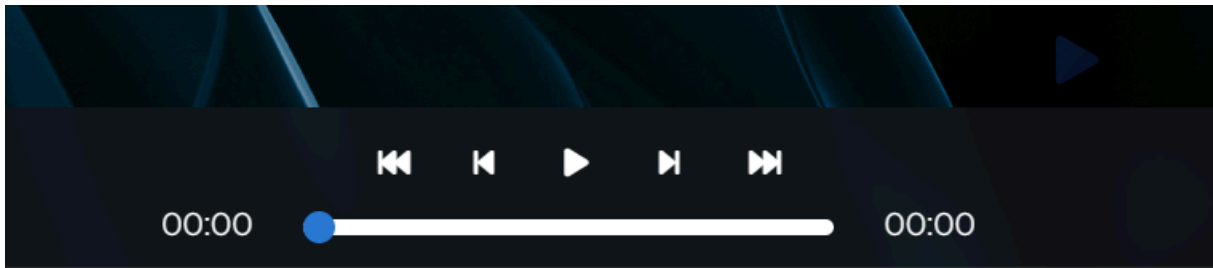
Zenét 5mp-el visszatekerése

Zene indítása/leállítása

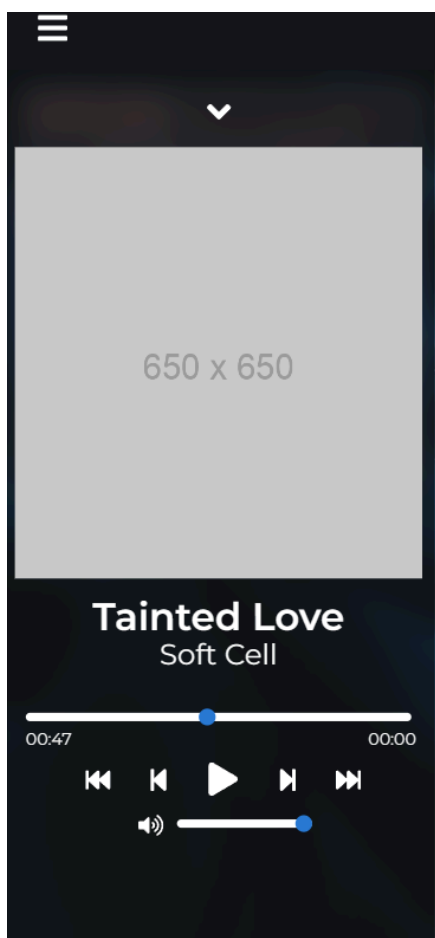
Zene 15mp előre léptetése

Zene átugrása

Ez alatt található hogy a felhasználó a zenében jelenleg hanyadik másodpercnél jár, ettől jobbra pedig egy sávval vizuálisan is megjelenítve, ha a felhasználó erre rákattint a zenét oda fogja ugrasztani. Ettől megint jobbra hogy mennyire hosszú a zene. A sávnak a jobboldalán található a némító gomb és a hangerő szabályzó. A némító gomb ikonja vizuálisan is jelzi a zenelejátszó mennyire hangos.



Ha a felhasználó telefonon nyitja meg akkor az alsó sávon a baloldali elemek változatlanok maradnak, viszont a jobboldalon már csak a zene elindítását/leállítására való gomb található. A többi elem eléréséhez a zenesávra kattintva érhető el. Ahol megtalálható az összes gomb amely a számítógépen is megtalálható



6.3. Navigációs menü

A navigációs menü tartalma változik attól függően hogy a felhasználó bevan-e jelentkezve illetve van-e admin felhasználói jogosultsága

6.3.1. Kezdőlap

A "Home"-ra kattintva lehet elérni a kezdőlapot. Ezen az oldalon található az összes elérhető zene

6.3.2. Kereső

A "Search"-re kattintva lehet elérni a keresőoldalt. Az oldalon zenékre lehet keresni, amelyekre rákattintva lehet elindítani a zenét

6.3.3. Lejátszási lista

A lejátszási listát csak bejelentkezett felhasználók tekinthetik meg, mivel magukhoz a felhasználóhoz vannak hozzákötve az általuk létrehozott lejátszási lista, tehát a felhasználók egymás lejátszási listáit nem tekinthetik meg. A zenelistákat a "Playlist" menüponttal lehet elérni. Felül egyszerűen lehet létrehozni lejátszási listát. A "New playlist name"-be beírja a felhasználó a lejátszási lista nevét és a jobboldalon rákattint a "Create Playlist"-re, amely azonnal megjelenik. A lejátszási lista nevére kattintva a rákattintott lejátszási lista oldalára kerül. Felül megtalálható a lejátszási lista neve és művelet gombok. A "Play"-re kattintva elindul a lejátszási listában található zenék. A "Modify"-ra kattintva megváltoztatható a lejátszási lista neve, a "Delete"-el pedig törölhető a lejátszási lista. Ez a menüsor alatt található meg az összes lejátszási listában található zene. Amik törölhetők is a "Delete" gombbal. Ettől a listától lentebb található az összes zene amiket az "Add" gombbal hozzáadható a lejátszási listához

6.4. Saját Profil

A "My Profile" menüpontra kattintva érhető el a felhasználó oldala. Felül egy szöveg köszönti a felhasználót. Ebben a menüben érhető el a felhasználó beállításának megváltoztatása, a profil neve és jelszava

6.4.1. Jelszó módosítása

A "Change password"-re kattintva az oldal átirányítja a felhasználót a "Reset Passowrd" oldalra, ahol először meg kell adnia a jelenleg jelszót és a kívánt jelszót, amelyet szintén meg kell erősíteni. A "Change password"-el elmenti a változtatásokat, és a felhasználó jelszava módosult. Az új jelszói szintén vonatkoznak a jelszó formai szabályai: A jelszónak minimum 8 karaktert kell tartalmaznia, illetve tartalmaznia kell Kis- és nagybetűt, számot, és egy speciális karaktert is

6.4.2. Profilnév módosítása

A "Change username" menüponttal a weboldal átirányítja a felhasználót a "Change username" oldalra ahol ha a felhasználó beírja a kívánt új felhasználó nevet.

6.5. Admin panel

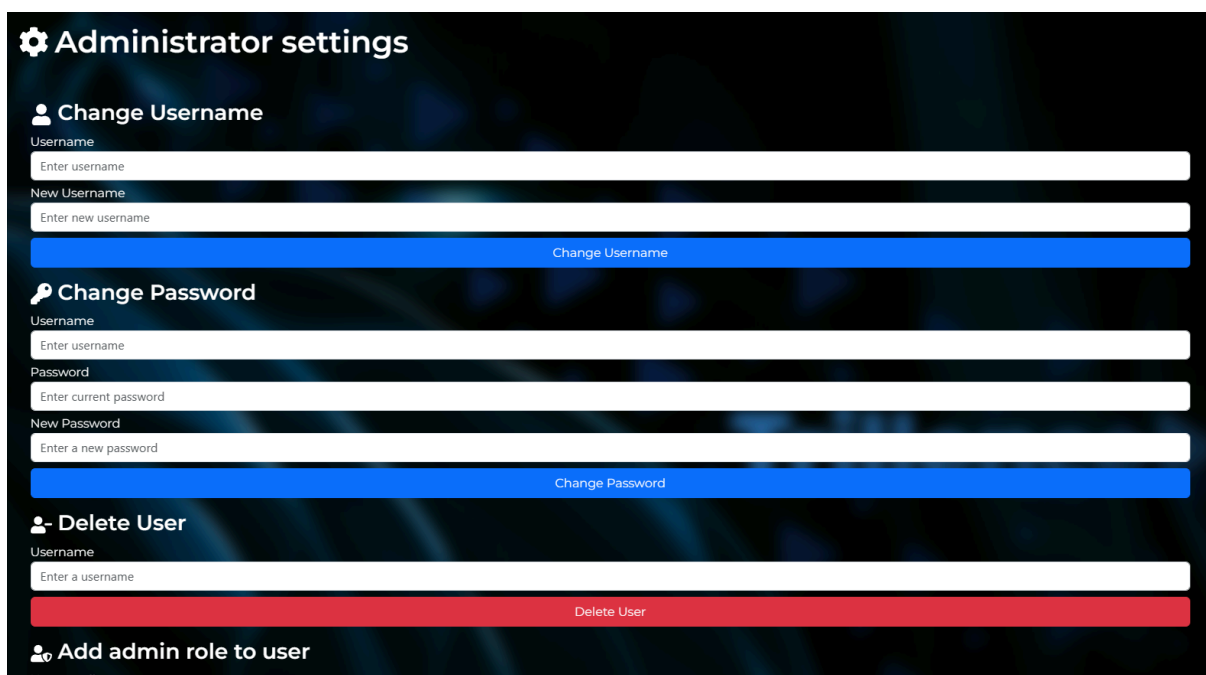
Az adminisztrátori panelt csak az adminisztrátori jogosultsággal rendelkező felhasználók számára elérhető.

Az első menüpontban Megváltoztatható a megadott felhasználónak, a felhasználó neve.

Az azt követő menüpontban megváltoztatható egy felhasználónak a jelszava amelyet itt is meg kell erősíteni.

Még lentebb törölhető a megadott felhasználó a felhasználónév megadásával

Az oldal legalján pedig adminisztrátori jogosultságot ad annak a felhasználónak amely a megadott email-el regisztrált



The screenshot displays the 'Administrator settings' interface. It features four main sections: 'Change Username' with input fields for 'Username' and 'New Username' and a blue 'Change Username' button; 'Change Password' with input fields for 'Username', 'Password' (current), and 'New Password', and a blue 'Change Password' button; 'Delete User' with an input field for 'Username' and a red 'Delete User' button; and 'Add admin role to user' with an input field for 'User email'.

6.6. SpotDL

Ez a menüpontot csak az adminisztrátor jogosultsággal rendelkező felhasználók számára elérhető. A projektbe integráltunk egy nyílt forráskódú projektet, amivel megkönnyebbült a zenék megfelelő formátumban való letöltése, és tesztelése.

6.6.1. SpotDL Használata

6.7. Bejelentkezés

A "Sign In" menüponttal érhető el a weboldal bejelentkező felülete ahol a felhasználó a beregisztrált felhasználó nevével és jelszavával jelentkezhet be. Amint a felhasználó bejelentkezett, onnantól elérhető számára a "Playlists" és a "My Profile" menüpontok. Ha a felhasználónak adminisztrátori jogai vannak, akkor elérhető lesz számára még a az "Admin panel" és a "SpotDI" menü.

6.7.1. Elfelejtett jelszó

Ha a felhasználó elfelejti a jelszavát, akkor a "Password" bemeneti mező alatt található a "Forgot password?" gombot, amelyre rákattintva átirányítja egy újabb

lapra, ahol a felhasználó email címét megadja, és a “reset passowrd” gombra kattintva a felhasználó kap az email címére egy “reset token”-t amelyet a “Reset token” bemeneti mezőbe bemásolva, és az új jelszót megadva, megváltoztathatja a jelszavát

6.8. Regisztráció

Ha a felhasználó be akarna regisztrálni akkor a “Sign in” menüpontban az oldal alján található a “Sign up” elemre kattintva elérheti a regisztrációs oldalt.

6.8.1. Regisztráció menete és követelményei

A regisztrációs oldalon a felhasználónak meg kell adnia egy felhasználó nevet (Username), az Email címét, és a jelszavát (Password), amelyet meg is kell erősítenie. A jelszónak minimum 8 karaktert kell tartalmaznia, illetve tartalmaznia kell Kis- és nagybetűt, számot, és egy speciális karaktert is

6.8.2. E-mail megerősítés

Amint a felhasználó rákattint a “Sign up” gombra, a megadott email címre amelyet a felhasználó megadott érkezik egy email amely tartalmazza a megerősítő kódot

6.9. Kijelentkezés

A “Sign Out” gombra kattintva a felhasználó kijelentkezik és a főoldalon landol

7. Backend

A backenden futó folyamatokat és végpontok elérésében a swagger segített:

7.1. authorization végpont:

7.1.1. register

Egy post metódus amely a felhasználó nevét, jelszavát, emailjét megkapva legenerál egy "register token" amelyet elküld a kért email címre

7.1.2. AssignRole

Egy post metódus amely az email címet és a jogosultságot megkapva megadja a megadott felhasználó jogosultságát

7.1.3. Login

Egy post metódus, mely a felhasználónevet és jelszót megkapva engedélyezi a bejelentkezést.

7.1.4. Change Username

Egy put metódus, mely a régi és új felhasználónevet megkapva lehetővé teszi a felhasználónév módosítását.

7.1.5. Change Password

Egy put metódus, mely a felhasználónevet, régi és új jelszót megkapva lehetővé teszi a jelszó módosítását.

7.1.6. send-reset-email

Egy post metódus, mely az email címet megkapva lehetővé teszi a jelszó visszaállításához szükséges visszaállító email elküldését.

7.1.7. reset-password

Egy put metódus, mely az email címet, új jelszót és visszaállítási tokent megkapva lehetővé teszi a jelszó visszaállítását.

7.1.8. delete-user

Egy delete metódus, mely a felhasználónevet megkapva lehetővé teszi a felhasználó törlését.

7.1.9. is-admin

Egy post metódus, mely a felhasználónevet megkapva lehetővé teszi ellenőrizni, hogy az adott felhasználó adminisztrátor-e. A frontend ezzel dönti el hogy mely navigációs menük jelenjenek meg

Auth

<https://localhost:7172/swagger/v1/swagger.json>

Auth



POST	/auth/register	✓
POST	/auth/AssignRole	✓
POST	/auth/login	✓
PUT	/auth/change-username	✓
PUT	/auth/change-password	✓
POST	/auth/send-reset-email	✓
PUT	/auth/reset-password	✓
DELETE	/auth/delete-user/{username}	✓
POST	/auth/is-admin	✓

7.2. Backend végpont:

7.2.1. General CRUD végpontok

Az adatbázisban lévő összes táblára készült végpont, de mivel ezek nagyrészt ugyanazok ezért ezeket nem íránk le részletesebben, de felsoroljuk a létező végpontokat:

Album

Artist

ArtistSong

Playlist

PlaylistSong

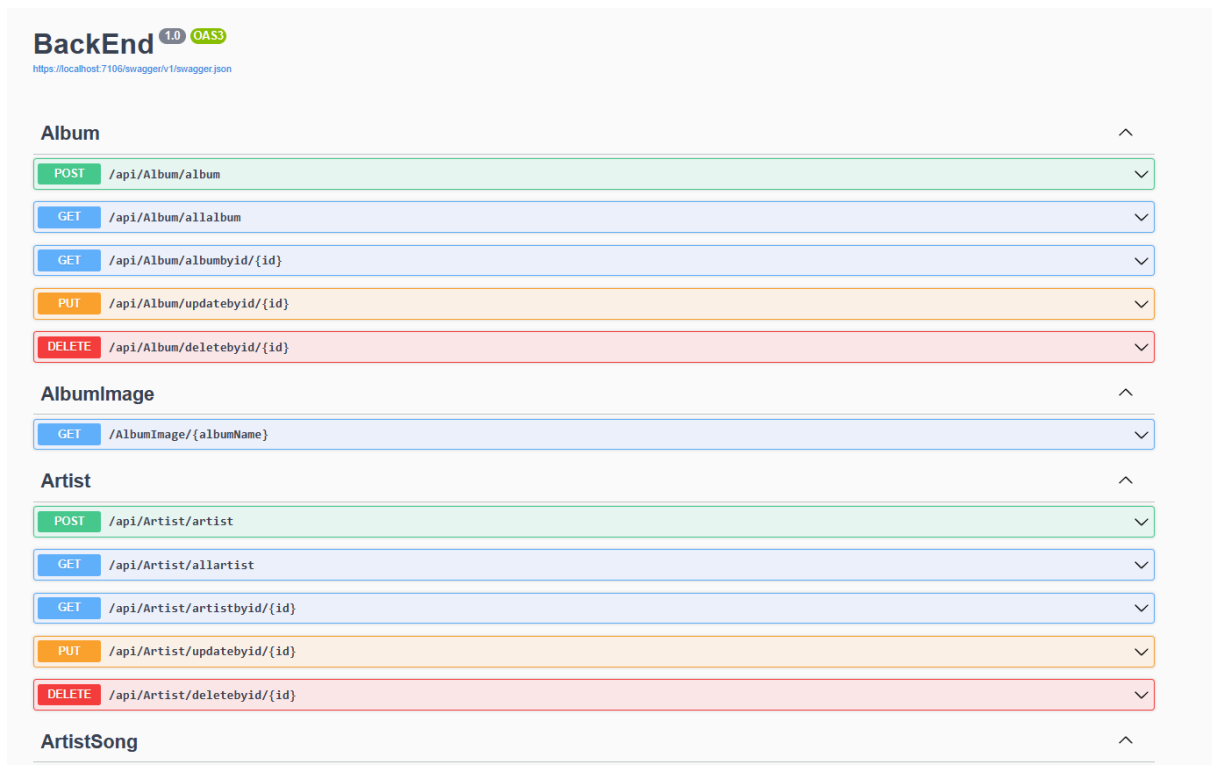
Song

User

Album	^
POST /api/Album/album	✓
GET /api/Album/allalbum	✓
GET /api/Album/albumbyid/{id}	✓
PUT /api/Album/updatebyid/{id}	✓
DELETE /api/Album/deletebyid/{id}	✓
AlbumImage	✓
Artist	^
POST /api/Artist/artist	✓
GET /api/Artist/allartist	✓
GET /api/Artist/artistbyid/{id}	✓
PUT /api/Artist/updatebyid/{id}	✓
DELETE /api/Artist/deletebyid/{id}	✓
ArtistSong	^
POST /api/ArtistSong/artist-song	✓
GET /api/ArtistSong/allartist-song	✓
GET /api/ArtistSong/artist-songbyid/{artistid, songid}	✓
PUT /api/ArtistSong/updatebyid/{artistid}/{songid}	✓
DELETE /api/ArtistSong/deletebyid/{artistid, songid}	✓
Connection	✓
Metadata	✓
Music Streaming	✓
Playlist	^
POST /api/Playlist/playlist	✓
GET /api/Playlist/allplaylist	✓
GET /api/Playlist/playlistbyid/{id}	✓
PUT /api/Playlist/updatebyid/{id}	✓
DELETE /api/Playlist/deletebyid/{id}	✓
Playlistsong	^
POST /api/Playlistsong/playlist-song	✓
GET /api/Playlistsong/allplaylistsong	✓
GET /api/Playlistsong/playlist-songbyid/{playlistid, songid}	✓
PUT /api/Playlistsong/updatebyid/{playlistid}/{songid}	✓
DELETE /api/Playlistsong/deletebyid/{playlistid}/{songid}	✓

7.2.2. Albumimage

Egy Get metódus amely segítségével lekéri a megadott mappából az összes albumképet



7.2.3. Connection

7.2.3.1. allsongdetails

Egy Get metódus, mely lekéri az összes albumot, az albumnak az előadóit, és végül az összes zenét és annak összes adatát

7.2.3.2. songdetailsbyid/Id

Egy Get metódus, mely lekéri a megadott zene neve alapján a megadott zene összes adatát

7.2.3.3. allplaylistdetails

Egy Get metódus, mely lekéri az összes playlistnek az összesadatát

7.2.3.4. allplaylistdetailsby/id

Egy Get metódus, mely lekéri a megadott playlistnek az összes adatát

7.2.4. Metadata

Ezt a metódust lefutattva az összes zenének a metadata automatikusa feltölti az adatbázisba Abból a mappából amelyet az appsettings.json-ben bevan állítva

7.2.5. Musicstreaming

7.2.5.1. current

Egy Get metódus, mely az alapértelmezett listából lejátssza az első számot

7.2.5.2. current/details

Egy Get metódus, mely lekéri a jelenleg lejátszott zenének az adatait

7.2.5.3. next

Egy Get metódus, mely a listát egyel nagyobb értéket ad

7.2.5.4. previous

Egy Get metódus, mely a listát egyel kisebb értéket ad

7.2.5.5. playlist/playlistid/current

Egy Get metódus, mely a megadott listából lejátssza az első számot

7.2.5.6. currentplaylist/playlistid/details

Egy Get metódus, mely lekéri a jelenleg lejátszott zenének az adatait

7.2.5.7. playlist/playlistid/next

Egy Get metódus, mely a listát egyel nagyobb értéket ad

7.2.5.8. playlist/playlistid/previous

Egy Get metódus, mely a listát egyel kisebb értéket ad

Connection		^
GET	/api/Connection/allsongdetails	▼
GET	/api/Connection/songdetailsbyid/{id}	▼
GET	/api/Connection/songdetailsbyname/{songName}	▼
GET	/api/Connection/allplaylistdetails	▼
GET	/api/Connection/playlistdetailsby/{id}	▼
Metadata		^
GET	/api/Metadata	▼
MusicStreaming		^
GET	/api/MusicStreaming/current	▼
GET	/api/MusicStreaming/current/details	▼
GET	/api/MusicStreaming/next	▼
GET	/api/MusicStreaming/previous	▼
GET	/api/MusicStreaming/playlist/{playlistId}/current	▼
GET	/api/MusicStreaming/currentplaylist/{playlistId}/details	▼
GET	/api/MusicStreaming/playlist/{playlistId}/next	▼
GET	/api/MusicStreaming/playlist/{playlistId}/previous	▼
GET	/api/MusicStreaming/stream	▼

8. Tesztelés

8.1.1. nUnit tesztek

8.1.2. ReactJS JEST tesztek

8.1.3. Lighthouse teljesítmény eredmények

9. Forrásjegyzék

<https://learn.microsoft.com/hu-hu/visualstudio/windows/?view=vs-2022&preserve-view=true>
https://learn.microsoft.com/en-us/aspnet/core/web-api/?view=aspnetcore-8.0&WT.mc_id=dotnet-35129-website
<https://www.w3schools.com/asp/default.ASP>
<https://react.dev/learn>
<https://www.w3schools.com/react/default.asp>
<https://typst.app/docs/>
<https://docs.npmjs.com/about-npm>
<https://getbootstrap.com/docs/5.3/getting-started/introduction/>