**Debreceni SZC Baross Gábor Technikum, Szakképző Iskola és Kollégium**

**Debrecen, 4030**

**Budai Ézsaiás u. 8/A.**

**A szakképesítés azonosító Száma:** **506131203**

## Szakképesítés megnevezése: Szoftverfejlesztő és -tesztelő

**Záródolgozat címe:**

**Magyarország eseményei mobil app és weblap fejlesztése**

**Záródolgozat készítője:**

**Csurgó Martin**

**Társam a projektben: Király Szabolcs**

**Konzulens: Dr. Várbíróné Nahaji Anikó**

**Konzultációs időpontok:**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Debrecen, 2025.**

Tartalomjegyzék

[Bevezetés 3](#_Toc32482713)

[Felhasználói dokumentáció 4](#_Toc32482714)

[Feladatspecifikáció 4](#_Toc32482715)

[Rendszerkövetelmények 4](#_Toc32482716)

[Program telepítése, elérése 4](#_Toc32482717)

[Program használatának leírása 4](#_Toc32482718)

[Fejlesztői dokumentáció 6](#_Toc32482719)

[Témaválasztás indoklása 6](#_Toc32482720)

[Alkalmazott fejlesztői eszközök 6](#_Toc32482721)

[Rendszerterv 6](#_Toc32482722)

[Adatmodell leírása 6](#_Toc32482723)

[Program leírása 6](#_Toc32482724)

[Részletes feladatspecifikáció, algoritmusok 6](#_Toc32482725)

[Tesztelés 7](#_Toc32482726)

[Továbbfejlesztési lehetőségek 7](#_Toc32482727)

[Összegzés 7](#_Toc32482728)

[Irodalomjegyzék 7](#_Toc32482729)

[Eredetiségnyilatkozat 8](#_Toc32482730)

# Bevezetés

A záródolgozatom egy olyan applikáció fejlesztéséről szól, amely lehetőséget ad arra, hogy Magyarország különböző eseményei között böngésszünk. Az applikáció segítségével könnyedén hozzáférhetünk az eseményekhez, és egyszerű szűrőeszközök révén az érdeklődésünknek megfelelő programokat találhatunk. Ez az applikáció fontos számomra, mivel szeretném jobban megismerni Magyarországot, és egyes események segítségével akár új helyeket is felfedezhetek. A téma választása mögött tehát egy erős személyes motiváció áll, hiszen a mobilalkalmazás lehetőséget ad arra, hogy a különböző rendezvények nyújtotta élményeket könnyen elérjem.

Az applikáció egy egyszerű, de rendkívül hasznos funkciókban gazdag rendszert kínál, amely lehetővé teszi az események közötti böngészést, azok szűrését és keresését. A felhasználók kétféleképpen érhetik el a tartalmat: vendégként vagy regisztrált felhasználóként. Vendégként bárki számára elérhető a rendszer, így az applikáció alapvető funkcióit – a keresést és a böngészést – bárki igénybe veheti. Regisztrált felhasználók számára viszont egy új lehetőség nyílik: az applikáció lehetőséget ad arra, hogy az érdeklődő eseményeket elmentsük, és egy külön fülön tekintsük meg azokat. Továbbá, a rendszer értesítést küld a felhasználónak egy órával az esemény kezdete előtt, így nem kell aggódniuk, hogy lemaradnak a számukra fontos rendezvényekről.

Az applikáció célja, hogy egy olyan platformot biztosítson, ahol bárki gyorsan és kényelmesen tájékozódhat a környéken zajló eseményekről. A funkciók és a felhasználói élmény szoros összhangban állnak azzal az igénnyel, hogy a látogatók számára lehetőséget adjon a felfedezésre, a közösségi eseményekre való figyelemfelkeltésre, és végül arra, hogy Magyarországot még jobban megismerjék és szeressék. Az applikáció tehát nemcsak egy praktikus eszközként szolgál, hanem segít a helyi kultúra és közösségek felfedezésében is.

# Felhasználói dokumentáció

## Feladatspecifikáció

A mobilalkalmazás célja, hogy a felhasználók böngészhessenek Magyarország eseményei között, valamint lehetőséget kapjanak saját események feltöltésére és megosztására.

Felhasználói szerepkörök:

* **Nem bejelentkezett felhasználó:**
  + Nyilvános események böngészése
  + Alkalmazás alapfunkcióinak megtekintése
* **Bejelentkezett felhasználó:**
  + Saját események feltöltése
  + Regisztrált események külön fülön való megtekintése
* **Admin:**
  + Felhasználói események moderálása (elfogadás/elutasítás)
  + Események törlése

**A mobil applikáció funkciói:**

* Regisztráció és bejelentkezés
* Események keresése és böngészése
* Saját események feltöltése
* Adminisztrációs felület az események kezelésére
* Kapcsolatfelvételi lehetőség

**Bemenő és kimenő adatok:**

* **Bemenő adatok:**
  + Regisztrációkor: e-mail, jelszó, felhasználónév
  + Esemény létrehozásakor: cím, leírás, dátum, helyszín, kategória, kép (opcionális)
  + Üzenet küldésekor: név, e-mail, üzenet szövege
* **Kimenő adatok:**
  + Publikus események listája
  + Saját események megjelenítése a külön fülön
  + Admin által elfogadott események

## Rendszerkövetelmények

#### **Hardverigények**

* **Eszközök:** Okostelefon, tablet
* **Minimális követelmények:**
* **Processzor:** Legalább egy középkategóriás ARM-alapú processzor (pl. Snapdragon 665 vagy Apple A10 és újabb)
* **RAM:** Minimum 2 GB ajánlott (jobb élményhez 4 GB vagy több)
* **Tárhely:** Az alkalmazás és a gyorsítótár számára legalább 100 MB szabad hely
* **Kijelző:** Legalább 720p felbontás, ajánlott Full HD vagy nagyobb

#### **Hálózati igények**

* Internetkapcsolat szükséges az események böngészéséhez, feltöltéséhez és a kapcsolatfelvételhez
* A mobilalkalmazás stabil működéséhez legalább 3G, de ajánlott a 4G vagy Wi-Fi kapcsolat

#### **Szoftverigények**

* **Operációs rendszerek:**
* iOS: iOS 12 vagy újabb
* Android: Android 8.0 (Oreo) vagy újabb

#### **Reszponzivitás és kijelző támogatás**

* **Mobiltelefonokra és tabletekre optimalizálva**
* **Ajánlott felbontás:** 1080×1920 (Full HD), de skálázódik kisebb és nagyobb kijelzőkhöz
* **Adaptív dizájn:** A kezelőfelület dinamikusan igazodik a különböző képernyőméretekhez

## Program telepítése, elérése

https://github.com/csurgomartinka/zarodoga\_backend

## 

## Program használatának leírása

### **HomeScreen**

Amikor megnyílik az app, ez az oldal jelenik meg. Itt három opció áll rendelkezésre:

* **Bejelentkezés**
* **Regisztráció**
* **Folytatás vendégként**

### **Főmenü**

A **Folytatás vendégként** gombra kattintva ez az oldal jelenik meg.

* Itt található rövid információ az alkalmazásról.
* Alul egy menüsorban elérhető a **Kapcsolat** és **Események** fül.
* Az oldal tartalmazza a főoldal és a kapcsolat fül tartalmát.

### **Események**

Itt regisztráció nélkül meg lehet tekinteni az eseményeket.

* **Kereső:** Az események neve alapján lehet keresni.
* **Szűrők:** Két legördülő menüből lehet szűrni eseménytípus és város szerint.
* Egy eseményre kattintva az alkalmazás átdob az **Esemény Oldala** füre.

### **Esemény Oldala**

Ha egy eseményre kattintunk, átlépünk erre az oldalra.

* Részletes leírást találunk az eseményről.
* Egyéb fontos információkat is megjelenít (időpont, helyszín stb.).

### **Regisztráció**

A **Regisztráció** gombra kattintva erre az oldalra jutunk.

* **Email ellenőrzés:** Figyeli, hogy tartalmazza-e az **@** jelet és utána egy domain-t.
* **Jelszó ellenőrzés:** Legalább 5 karakter hosszú, tartalmaz egy nagybetűt és egy számot.
* Sikeres regisztráció esetén átdob a **Bejelentkezés** oldalra.

### **Bejelentkezés**

A bejelentkezés gombra kattintva ez az oldal jelenik meg.

* Be lehet lépni egy már regisztrált email és jelszó párossal.
* Az email és jelszó megfelelőségét ellenőrzi.
* **Elfelejtett jelszó** gomb segít a jelszó visszaállításban.
* Sikeres bejelentkezés után a **Bejelentkezett** oldalra lép.

### **Elfelejtett Jelszó**

Az elfelejtett jelszó gombra kattintva ide jutunk.

* Itt egy regisztrált email címet kell megadni.
* Az alkalmazás továbbléptet a **Jelszókód Megadás** oldalra.

### **Jelszókód Megadása**

Az alkalmazás egy véletlenszerű kódot generál és elküldi a megadott email címre.

* Ha helyesen adjuk meg a kódot, továbblép az **Új jelszó beállítása** oldalra.

### **Új Jelszó Beállítása**

* Kétszer kell megadni az új jelszót.
* Ellenőrzi, hogy megfelel-e a regisztrálási feltételeknek, valamint hogy a két jelszó egyezik-e.

### **Bejelentkezett Felhasználó Főoldala**

* Kis információ rólunk és az appról.
* Alul menü: **Kapcsolat, Események, Profilom**

### **Bejelentkezett Felhasználó Események**

* Ugyanaz, mint az eseményoldal, de van:
  + **Kijelentkezés gomb**
  + **Felhasználók által létrehozott események gomb**, ami átdob az **Események Felhasználók** oldalra.

### **Bejelentkezett esemény részletei**

* Az esemény információi megjelennek.
* Két extra funkció:
  + **Megjelölheti, hogy az esemény érdekli-e.**
  + **Ha admin, törölheti az eseményt.**

### **Saját Események**

* A felhasználó által **érdekesnek jelölt események** és **létrehozott események** listázása.
* Gomb az eseménylétrehozáshoz.
* **Admin felület** az események elfogadására vagy elutasítására.

### **Esemény Létrehozás**

* A felhasználók itt hozhatnak létre eseményeket.
* Minden mező kitöltése kötelező.
* Saját város megadása is lehetséges.

### **Felhasználók által létrehozott események**

* Itt az admin által már jóváhagyott események jelennek meg.
* A bejelentkezett felhasználók böngészhetik ezeket.

### **Felhasználók által létrehozott események részletei**

* Egy felhasználó által létrehozott esemény részletei jelennek meg.
* **A tartalom megegyezik a többi eseményoldallal.**

# Fejlesztői dokumentáció

## Témaválasztás indoklása

### **Program rövid összefoglalása**

Az alkalmazás egy eseményböngésző platform Magyarországon, amely lehetővé teszi a felhasználók számára, hogy különböző eseményeket fedezzenek fel, saját eseményeket töltsenek fel és kapcsolatba lépjenek a szervezőkkel vagy az alkalmazás fejlesztőivel.

### **Miért ezt választottam?**

Azért döntöttem egy ilyen alkalmazás mellett, mert szeretném bejárni Magyarországot, és egy olyan eszköz sokat segíthet, amelyben előre meg tudom nézni az érdekes eseményeket. Ez motivál arra, hogy felfedezzek új helyeket és élményeket szerezzek.

### **Biztonsági kérdések**

**Hitelesítés és jogosultságkezelés:** A felhasználók bejelentkezése biztonságos módon történik, és az egyes funkciókhoz eltérő jogosultsági szintek tartoznak (pl. eseményfeltöltés csak bejelentkezett felhasználóknak, moderálás csak adminoknak).

**Adatvédelem**: A felhasználók adatai titkosított formában kerülnek tárolásra, és az eseményfeltöltésnél szűrők védik az alkalmazást a rosszindulatú tartalmaktól.

## Alkalmazott fejlesztői eszközök

#### **Fejlesztési eszközök és programozási nyelvek**

* **Mobilalkalmazás fejlesztése:** React Native
* **Adatbázis-kezelés:** phpMyAdmin
* **Adatkinyerés és háttérfolyamatok:** Python
* **Fejlesztői környezet:** Visual Studio Code

#### **Felhasznált modulok és források**

* [React Native TextInput](https://reactnative.dev/docs/textinput)
* [React Native Button](https://reactnative.dev/docs/button)
* [Expo Date-Time Picker](https://docs.expo.dev/versions/latest/sdk/date-time-picker/)
* [W3Schools JavaScript Fetch API](https://www.w3schools.com/js/js_api_fetch.asp)

## Rendszerterv

### Adatmodell leírása

**esemeny tábla:**

|  |  |
| --- | --- |
| **Id:** Elsődleges kulcs.  **nev**: Az események nevét tartalmazza.  **datum**: Szövegesen leírva az esemény dátumát tartalmazza.  **varosid:** Másodlagos kulcs. A varos tábla vnev oszlopára hivatkozik, ami a városok nevét tartalmazza.  **tipusid:** Másodlagos kulcs. A tipus tábla tipus\_nev oszlopára hivatkozik, ami az esemény típusát tartalmazza.  **leiras**: Az eseményről szóló rövid leírás, bemutatás.  **reszletek**: Az eseményről bővebb leírás, fontos részletek.  **datum\_kezdet**: Az eseménynek az első napjának dátumát tartalmazza.  **datum\_veg:** Az eseménynek az utolsó napjának dátumát tartalmazza. |  |

**erdekelt\_esemeny tábla**

Ezt a táblát a mobilos applikáció használja. Az események közül ki lehet választani azokat, amik minket érdekelnek és ezt egy külön fülön jeleníti meg az applikáció. Az események frissülése miatt az esemény nevét és dátumát tároltam el.

|  |  |
| --- | --- |
| **Id:** Elsődleges kulcs.  **Felhasznaloid:** A felhasznalo tábla id oszlopára hivatkozik, amivel a felhasználót lehet beazonosítani.  **Esemeny\_nev:** A felhasználó által érdekelt eseménynek a nevét tárolja.  **Esemeny\_datum:** A felhasználó által érdekelt eseménynek a kezdésének dátumát tárolja.  **torolt:** Azt vizsgálja, hogy az esemény törölve lett-e, eszerint visszajelzést küldünk a felhasználónak. |  |

**felhasznalo tábla**

|  |  |
| --- | --- |
| **Id:** Elsődleges kulcs.  **Email:** A regisztrált felhasználó email címét tárolja le.  **Password:** A regisztrált felhasználó jelszavát tárolja le titkosítva.  **Jogosultsag:** A regisztrált felhasználó hozzáférését szabályozza. |  |

**Felhasznalo\_esemeny tábla**

|  |  |
| --- | --- |
| **Id:** Elsődleges kulcs.  **Nev:** Az események nevét tartalmazza.  **Datum\_mettol**: Az eseménynek az első napjának dátumát tartalmazza.  **Datum\_meddig:** Az eseménynek az utolsó napjának dátumát tartalmazza.  **Varos:** Az esemény városát tartalmazza.  **Tipus:** Az esemény tipusát tartalmazza.  **Leiras:** Az eseményről szóló rövid leírás, bemutatás.  **Reszletek:** Az eseményről bővebb leírás, fontos részletek.  **Helyszin:** Az esemény helyének adatait tartalmazza.  **Felhasznalo:** Az eseményt létrehozó felhasználó email címét tartalmazza.  **Elfogadott:** Az esemény állapotát tartalmazza (0) elfogadásra vár. (1) elfogadott. (2) elutasított. |  |

**helyszin tábla**

|  |  |
| --- | --- |
| **Helyszin\_id:** Elsődleges kulcs.  **Helyszin\_nev:** Az esemény helyének adatait tartalmazza.  **Varosid:** Másodlagos kulcs. A varos tábla vnev oszlopára hivatkozik, ami a városok nevét tartalmazza  **Esemeny\_id:** Az esemeny tábla id oszlopára hivatkozik, amivel az eseményeket lehet beazonosítani. |  |

**kapcsolat tábla**

|  |  |
| --- | --- |
| **kapcsolat\_id:** Elsődleges kulcs.  **nev:** Az üzenetet küldő felhasználó nevét tartalmazza.  **email:** Az üzenetet küldő felhasználó email címét tartalmazza.  **uzenet:** A felhasználó által küldött üzenetet tartalmazza.  **datum:** Az üzenet küldésének dátumát tartalmazza.  **uzenet\_elfogadva:** Az üzenet állapotát tartalmazza (1) elfogadott, (0) elfogadásra vár. |  |

**tipus tábla**

|  |  |
| --- | --- |
| **tipus\_id:** Elsődleges kulcs.  **tipus\_nev:** Az esemény tipusát tartalmazza. |  |

**varos tábla**

|  |  |
| --- | --- |
| **id:** Elsődleges kulcs.  **vnev:** Az esemény városát tartalmazza. |  |

### Program leírása

használt php és egyéb js fájlok és feladataik leírása, hierarchia bemutatása rajzzal, kikopizhatod pl. commanderből, vscode-ból, hogy egy-egy mappádon belül milyen fájlok vannak, rövid leírás.

## Részletes feladatspecifikáció, algoritmusok

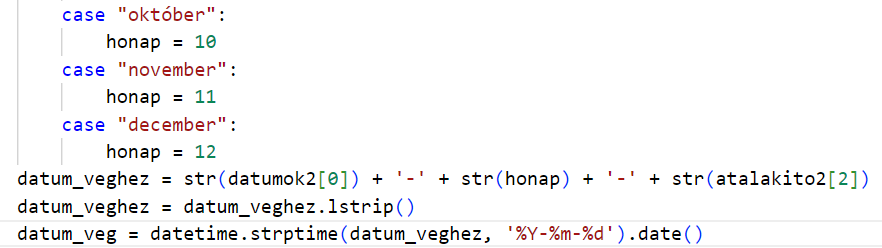
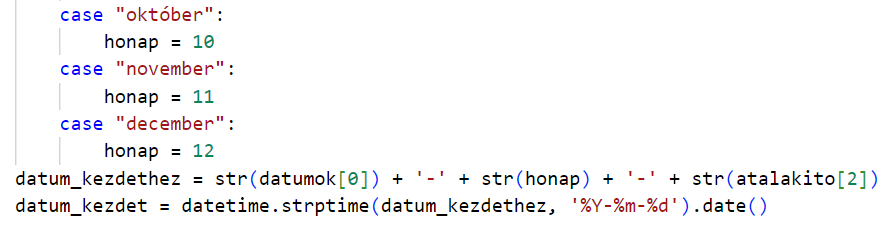
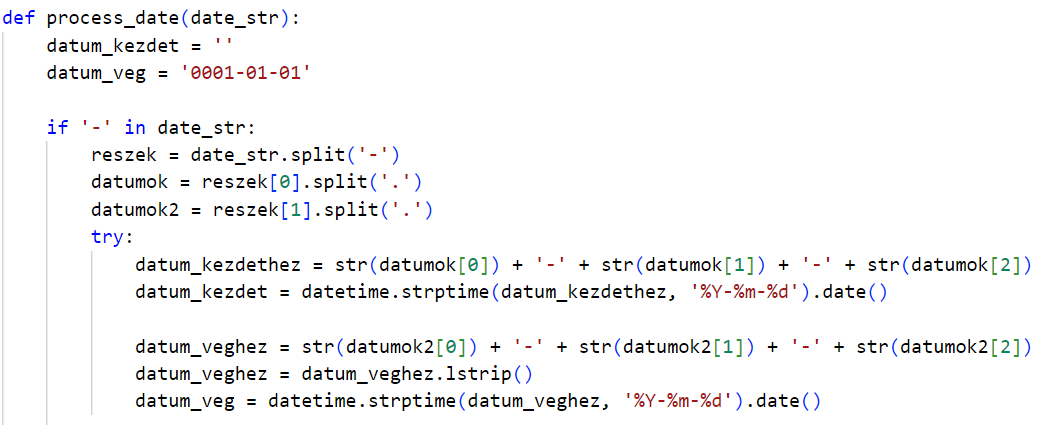
**Dátumformátumok intelligens feldolgozása (process\_date függvény)**

**Mappa / fájl:** zarodoga/teszt.py

**Leírás:** Az események dátumai különböző formátumokban érkeznek (pl. 2025.01.11., 2025 január 11. stb.), és ezeket a függvény egységes formátumra alakítja.

**Megoldás:**

* Az if '-' in date\_str: feltétellel ellenőrzi, hogy az adott dátum egy időszakot fed-e le.
* Ha az év, hónap és nap számjegyben szerepel, akkor datetime.strptime segítségével közvetlenül konvertálja.
* Ha a hónap szöveges formában van, akkor egy match-case szerkezet segítségével számértékre alakítja.



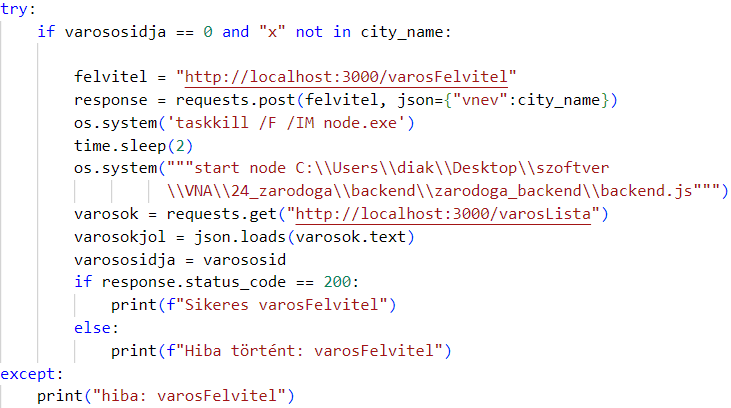
**Automatikus városfelvitel és backend újraindítás**

**Mappa / fájl:** zarodoga/teszt.py

**Leírás:** Ha egy új város kerül feldolgozásra, és nincs az adatbázisban, akkor automatikusan hozzáadja, majd újraindítja a backend szervert, hogy az adatok frissüljenek.

**Megoldás:**

* Ha egy város nincs az adatbázisban, akkor requests.post() segítségével felküldi az API-nak.
* Az os.system('taskkill /F /IM node.exe') paranccsal leállítja a backendet.
* Ezután vár 2 másodpercet, hogy elkerülje az adatbázis-összeomlást (time.sleep(2)).
* Végül újraindítja a backend szervert (os.system('start node backend.js')).



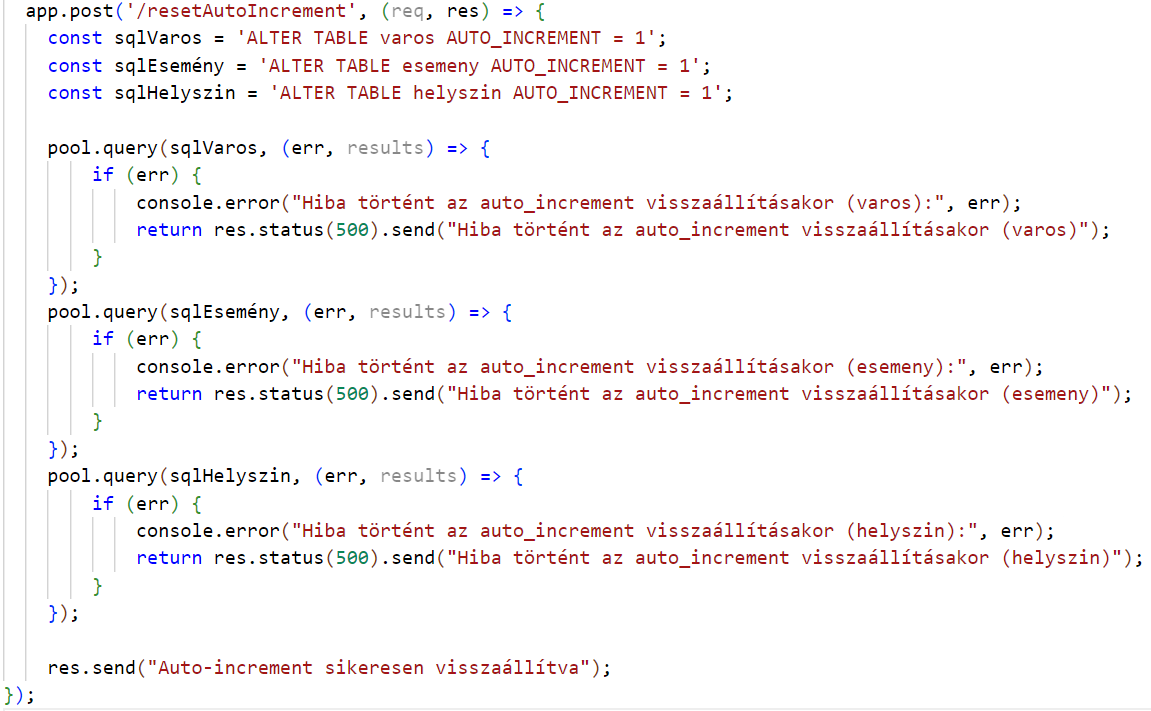
**AUTO\_INCREMENT visszaállítása minden indításnál**

**Mappa / fájl:** zarodoga\_backend/backend.js

**Leírás:** Az adatbázisban tárolt események és helyszínek id értéke folyamatosan növekedne, ha nem törölnénk a rekordokat rendszeresen. Ennek elkerülése érdekében minden egyes új indításkor visszaállítom az AUTO\_INCREMENT értékeket az alábbi módszerrel.

**Megoldás:**

* Az ALTER TABLE <tábla\_név> AUTO\_INCREMENT = 1 SQL parancs segítségével visszaállítom az elsődleges kulcs (id) számlálót.
* Az adatbázis pool segítségével párhuzamosan hajtom végre a lekérdezéseket.
* Hibaellenőrzés is be van építve, így ha egy visszaállítás sikertelen, az alkalmazás logolja az errort és egy HTTP 500-as hibakódot küld vissza.



## Továbbfejlesztési lehetőségek

Értesítések az esemény kezdete előtt:

* A felhasználók beállíthatnák, hogy egy adott időpontban (pl. 1 órával vagy 1 nappal korábban) értesítést kapjanak a megjelölt eseményekről.
* Opcionális beállítás lehetne az ismétlődő eseményeknél (pl. heti értesítés).

Térképes megjelenítés:

* Az események térképen is megjelenhetnének, így a felhasználók könnyebben tervezhetik az útvonalukat.
* Integráció Google Maps vagy OpenStreetMap alapon.

Közösségi funkciók:

* Kommentelési lehetőség az eseményeknél, hogy a felhasználók megoszthassák véleményüket, tapasztalataikat.
* Felhasználók követése – ha valakinek tetszenek egy másik felhasználó által szervezett események, követhetné az illetőt, hogy értesüljön az új eseményeiről.

# Összegzés

A dokumentációm egy olyan mobilalkalmazást mutat be, amely lehetővé teszi a felhasználók számára, hogy Magyarország eseményei között böngésszenek, valamint saját eseményeket töltsenek fel. A fejlesztés során számos különböző technológiát alkalmaztam, köztük **React Native, phpMyAdmin és Python**, amelyek segítségével egy jól működő rendszert hoztam létre.

A projekt **hasznossága több szempontból is kiemelkedő**:

* **Turisták és utazók számára**: Az alkalmazás egyértelműen segíti azokat, akik felfedeznék Magyarországot, hiszen könnyen és gyorsan találhatnak eseményeket az ország különböző pontjain.
* **Helyi szervezőknek és közösségeknek**: Az eseményfeltöltési lehetőség lehetővé teszi, hogy kisebb és nagyobb közösségek is hirdethessék programjaikat, így több emberhez juthatnak el.

### **Saját Munka Értékelése**

Úgy gondolom, hogy a szakdolgozatom és maga a fejlesztés is **jól sikerült**. A projekt során rengeteget tanultam:

* Az első félévben tanult **Python tudásomat** tovább mélyítettem, és sikeresen alkalmaztam a gyakorlatban.
* Megtanultam kezelni a **backend és frontend integrációját**, és hatékonyan használni az **adatbázisokat**.
* Több **kreatív megoldást** is alkalmaztam, például a szerver **újraindítását** annak érdekében, hogy a Python adatkérései mindig a friss adatokat érjék el.

Az alkalmazás fejlesztése során természetesen voltak kihívások, de ezek leküzdése még értékesebbé tette az egész folyamatot. Azt tapasztaltam, hogy az **önálló tanulás és problémamegoldás** kulcsfontosságú a fejlesztői munka során, és örülök, hogy ezt élesben is megtapasztalhattam.

Összességében büszke vagyok arra, amit elértem, hiszen egy működőképes és hasznos alkalmazást készítettem, amely valós problémákra nyújt megoldást. Ez a projekt nemcsak szakmailag vitt előre, hanem egyben **megerősített abban, hogy a fejlesztés világa az én utam**.

# Irodalomjegyzék

### **Internetes Források**

#### **React Native és Expo dokumentációk:**

* Axios: <https://www.npmjs.com/package/react-native-axios>
* TextInput: <https://reactnative.dev/docs/textinput>
* TouchableOpacity: <https://reactnative.dev/docs/touchableopacity>
* Button: <https://reactnative.dev/docs/button>

#### **Webfejlesztés és programozás:**

* W3Schools: [https://www.w3schools.com](https://www.w3schools.com/)  
   (Ez a forrás alapvető HTML, CSS, JavaScript, SQL és PHP példákat biztosított.)

#### **Web Scraping Pythonban:**

* Beautiful Soup: <https://realpython.com/beautiful-soup-web-scraper-python/>  
   (Segítségével adatokat nyertem ki weboldalakról a Python scriptjeimhez.)

# Eredetiségnyilatkozat

Alulírott Saját Név nyilatkozom, hogy záró dolgozatom a saját szellemi termékem.

Debrecen, 2025. márc. 28.

.

aláírás