**2024 -2025**

**KaposTransit**

**Készítette: Falka Marietta & Bogdán László**Szoftverfejlesztő és tesztelő képzés

**Kaposvár**

## Kaposvári Szakképzési Centrum Noszlopy Gáspár Közgazdasági Technikum

## Tartalomjegyzék

1. **Bevezetés**
   1. A Projekt háttere
   2. Célkitűzéseink
   3. Jelenlegi megoldások hiányosságai
2. **Rendszerkövetelmények**
   1. Hardver követelmények
   2. Szoftver követelmények
   3. Hálózati követelmények
3. **Első lépések**
   1. Regisztrációs folyamat
   2. Bejelentkezési módok
4. **Fő funkciók**
   1. Menetrendek
   2. Célállomás kiválasztása
   3. Járatállapot figyelő
5. **GTFS integráció**
   1. GTFS formátum áttekintés
   2. Támogatott partnerek
6. **API dokumentáció**
   1. API áttekintés
   2. Végpontok

6.2.1 Menetrend végpontok

* 1. Példa kérések és válaszok

1. **Mobil kompatibilitás**
   1. Reszponzív dizájn
   2. Mobilspecifikus funkciók
2. **Biztonsági és adatvédelmi szabályzat**
   1. Felhasználói adatok védelme
   2. Adatkezelési tájékoztató
   3. GDPR megfelelőség
3. **Tesztelés**
   1. Automatizált tesztelés
   2. Manuális tesztelés
4. **Hibakeresés és támogatás**
   1. Gyakori hibajelenségek
   2. Ügyfélszolgálati elérhetőségek
   3. Hibabejelentési folyamat
5. **Fejlesztési tervek**
   1. Jövőbeni funkciók
   2. Felhasználói visszajelzések kezelése
   3. Mobil alkalmazás lehetőségek
   4. API bővítések
6. **Fejlesztési tervek**
   1. Bejelentkezési rendszer
   2. Irányítópult
   3. Késések kezelése
   4. Hírek kezelése
   5. Az admin felület használata
   6. Jövőbeli fejlesztések

13

.

**Melléklet**

1. x

13.1

Fogalomtár

13.2

Gyors hivatkozások

1. Bevezető

### 1.1 A Projekt háttere

A mindennapokban használt utazás nehézségeinek leküzdése.

A projekt célja a Kaposvári Közlekedési Zrt. helyi járatos közlekedésének könnyebb, átláthatóbb felhasználóbarát felületének létrehozása volt.

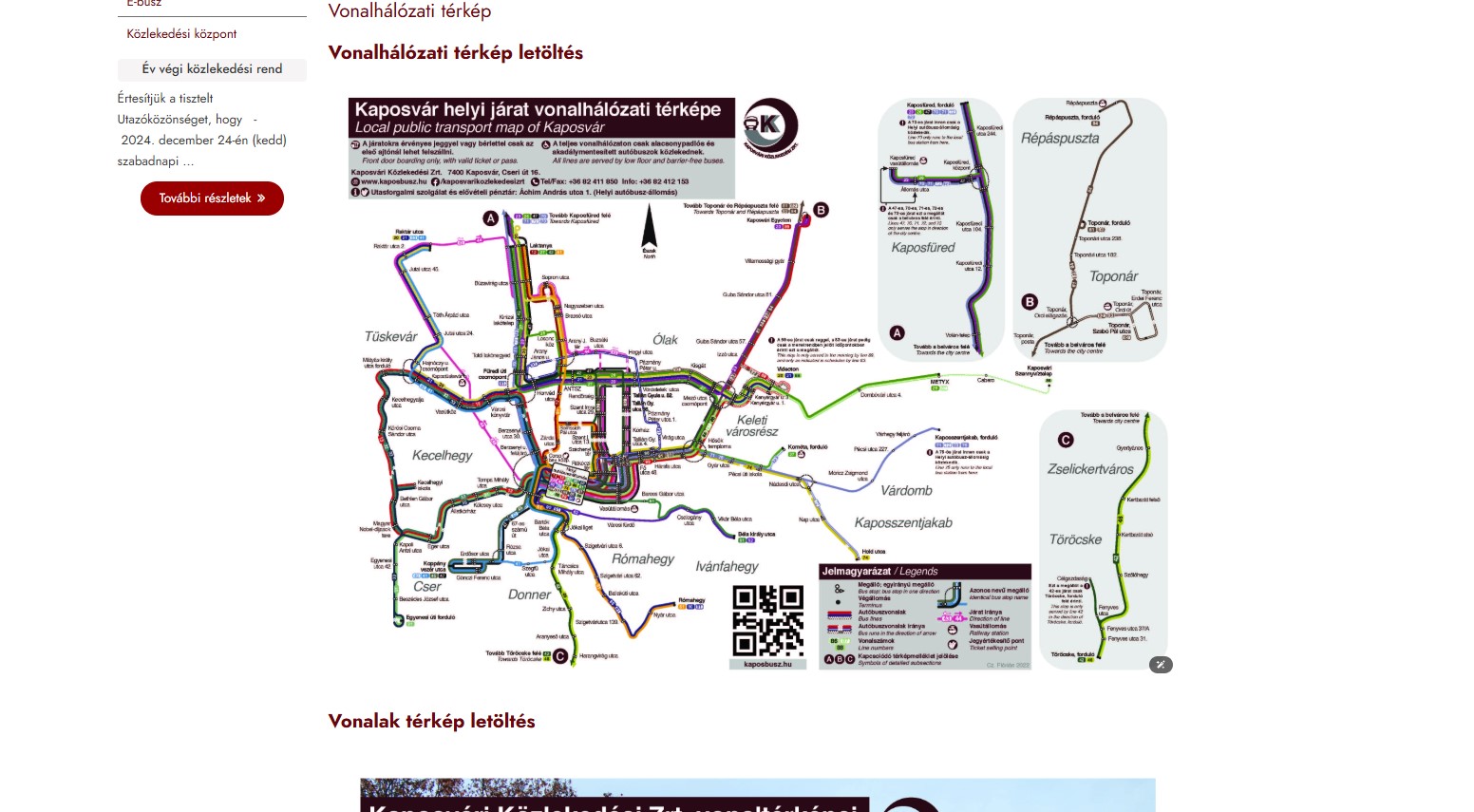
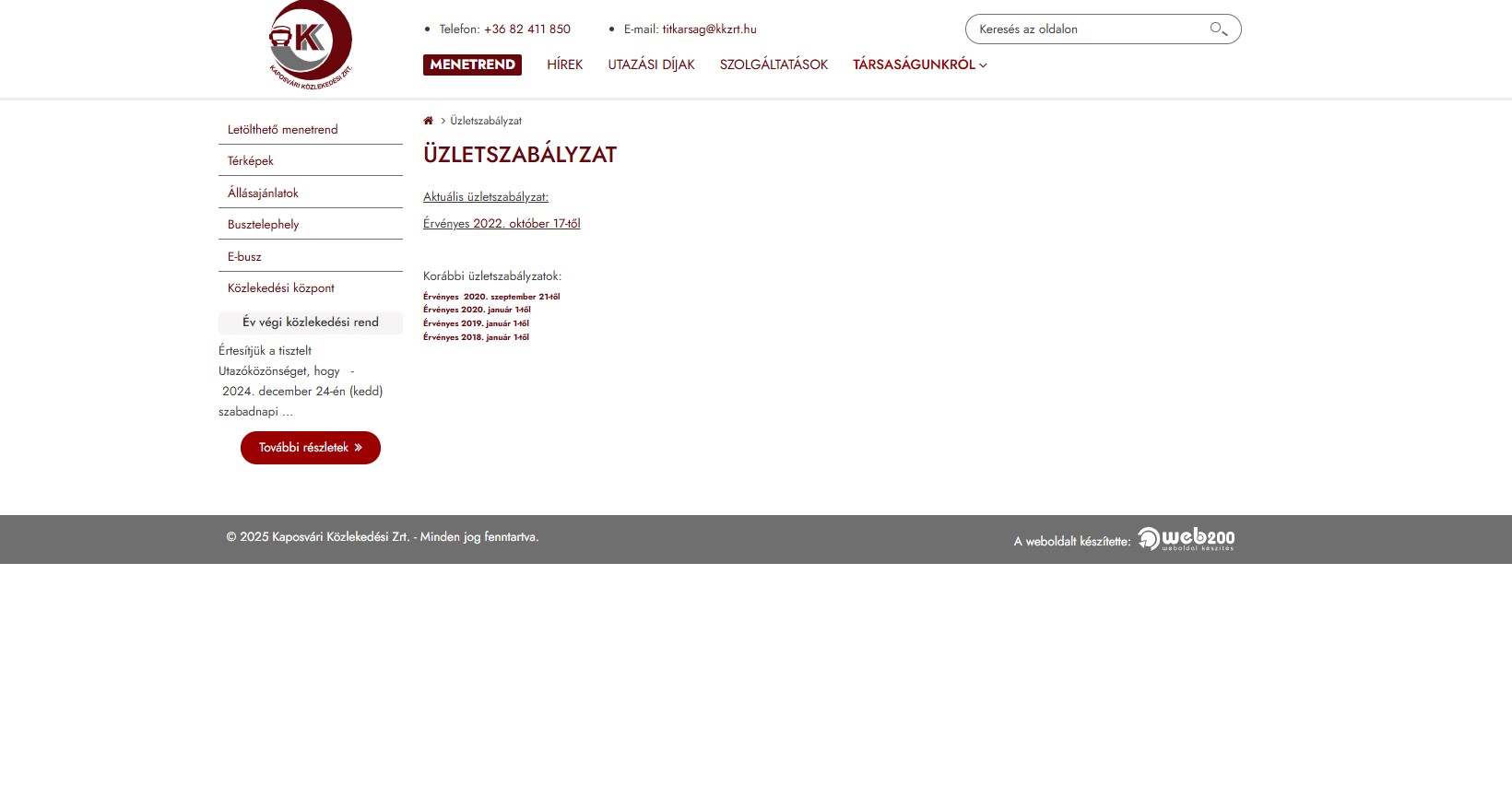
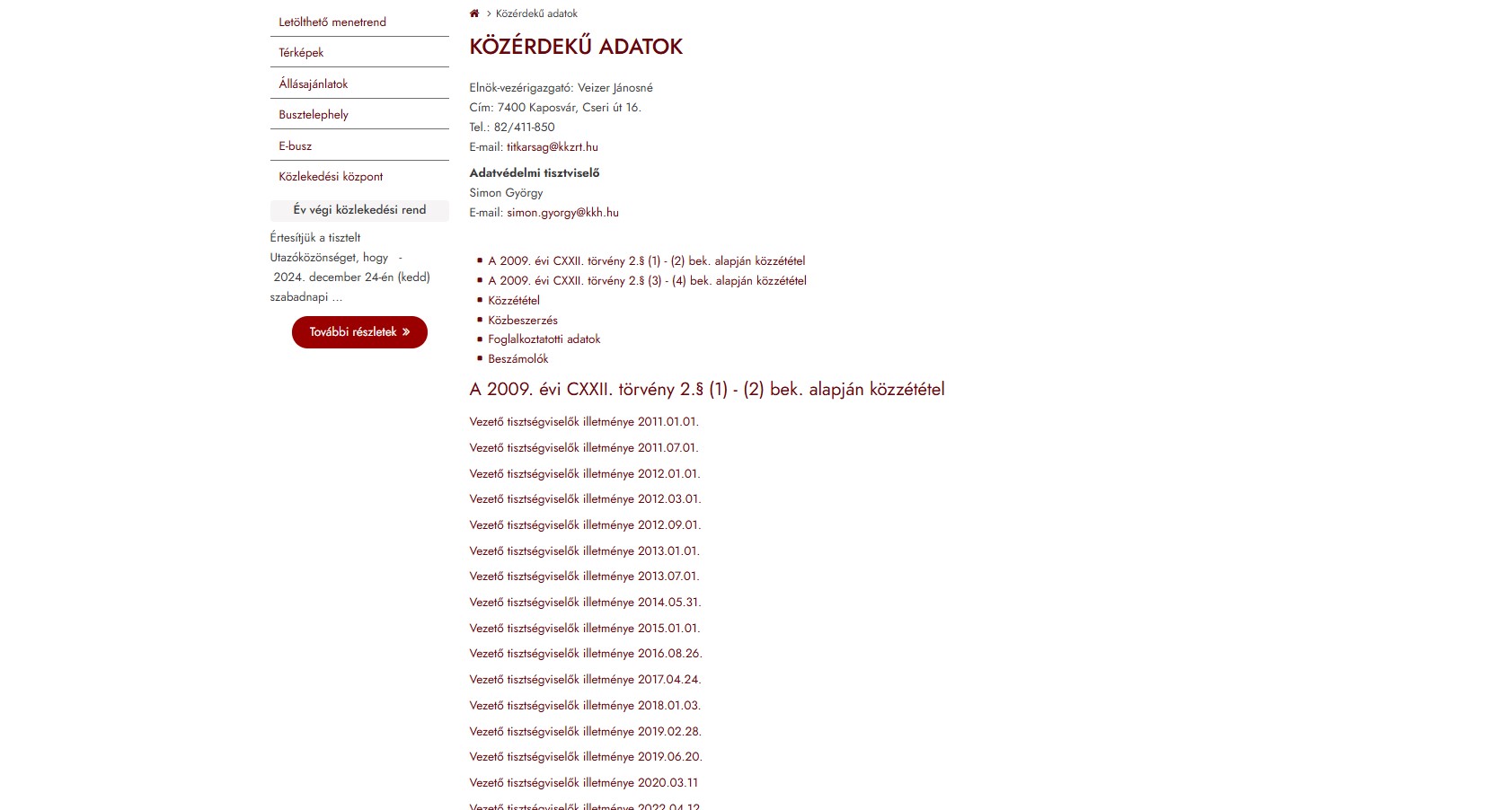
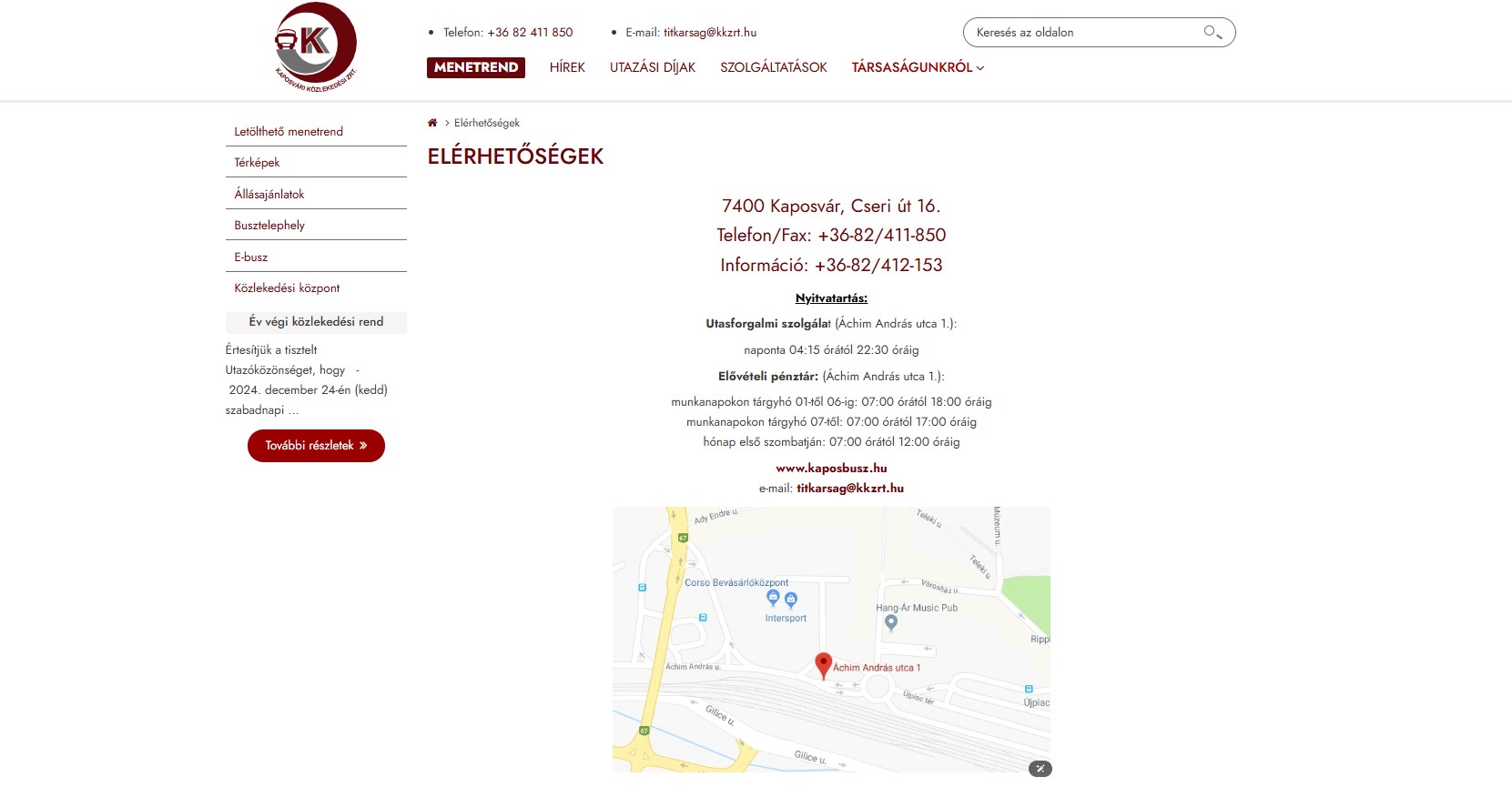
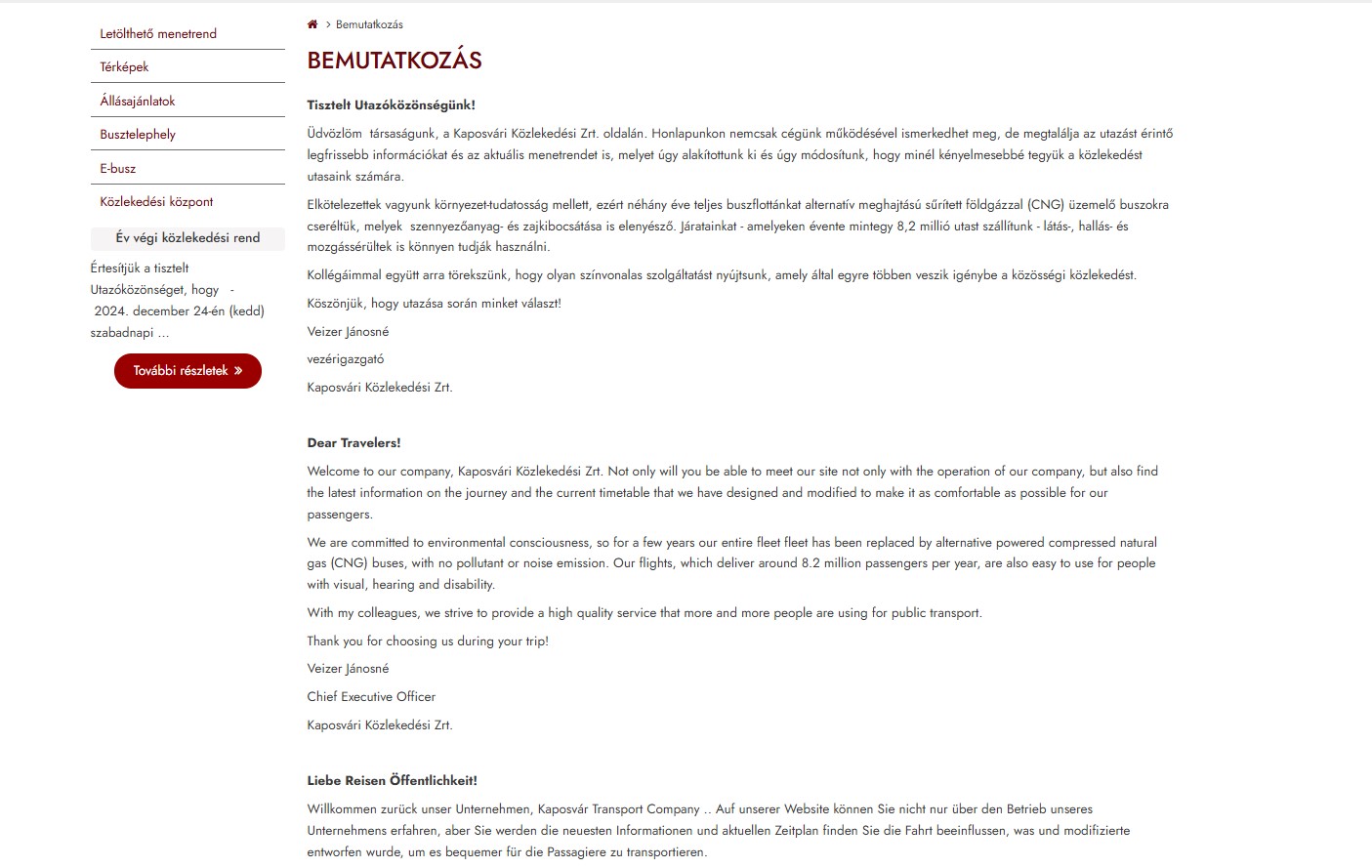
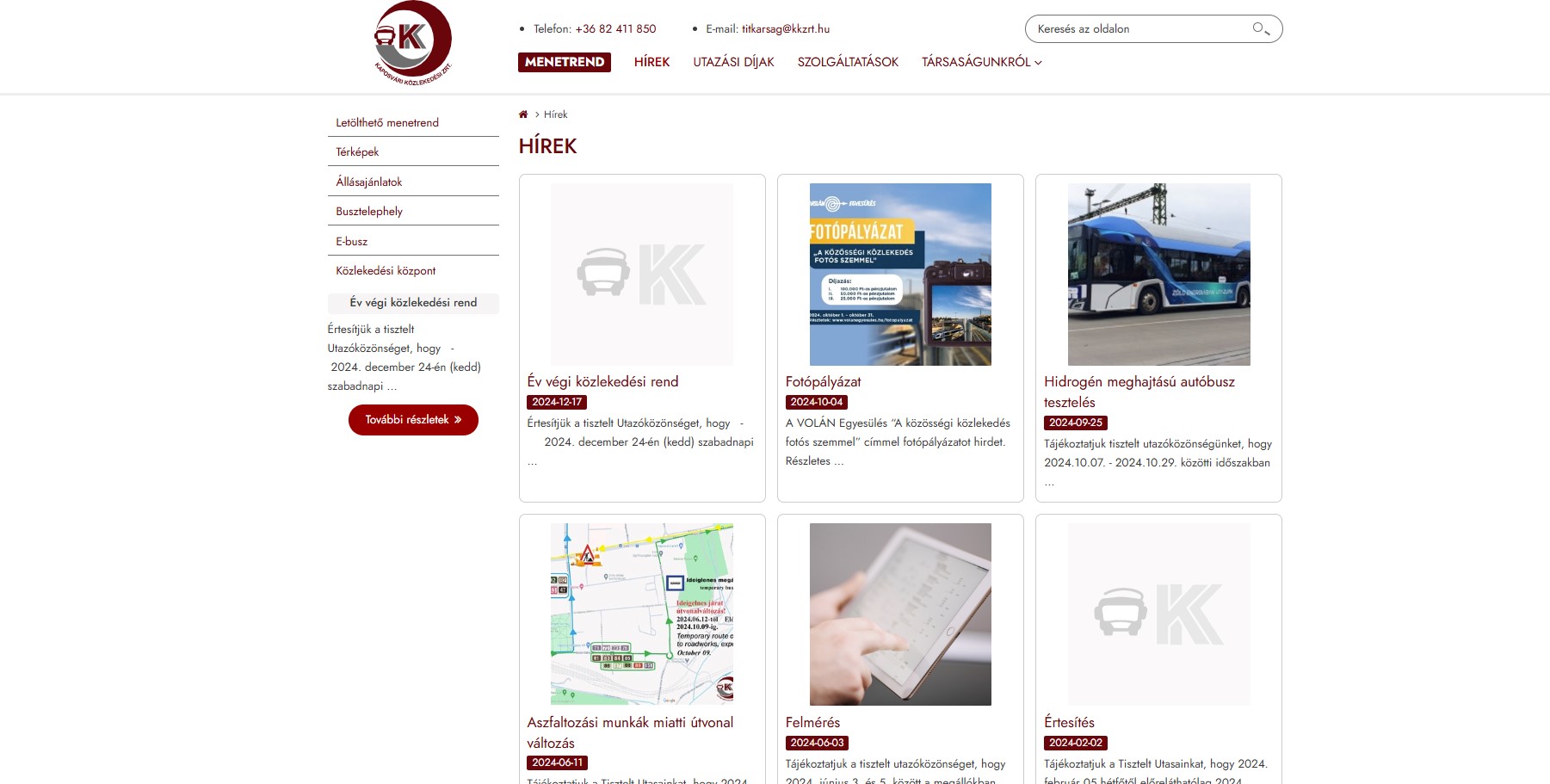
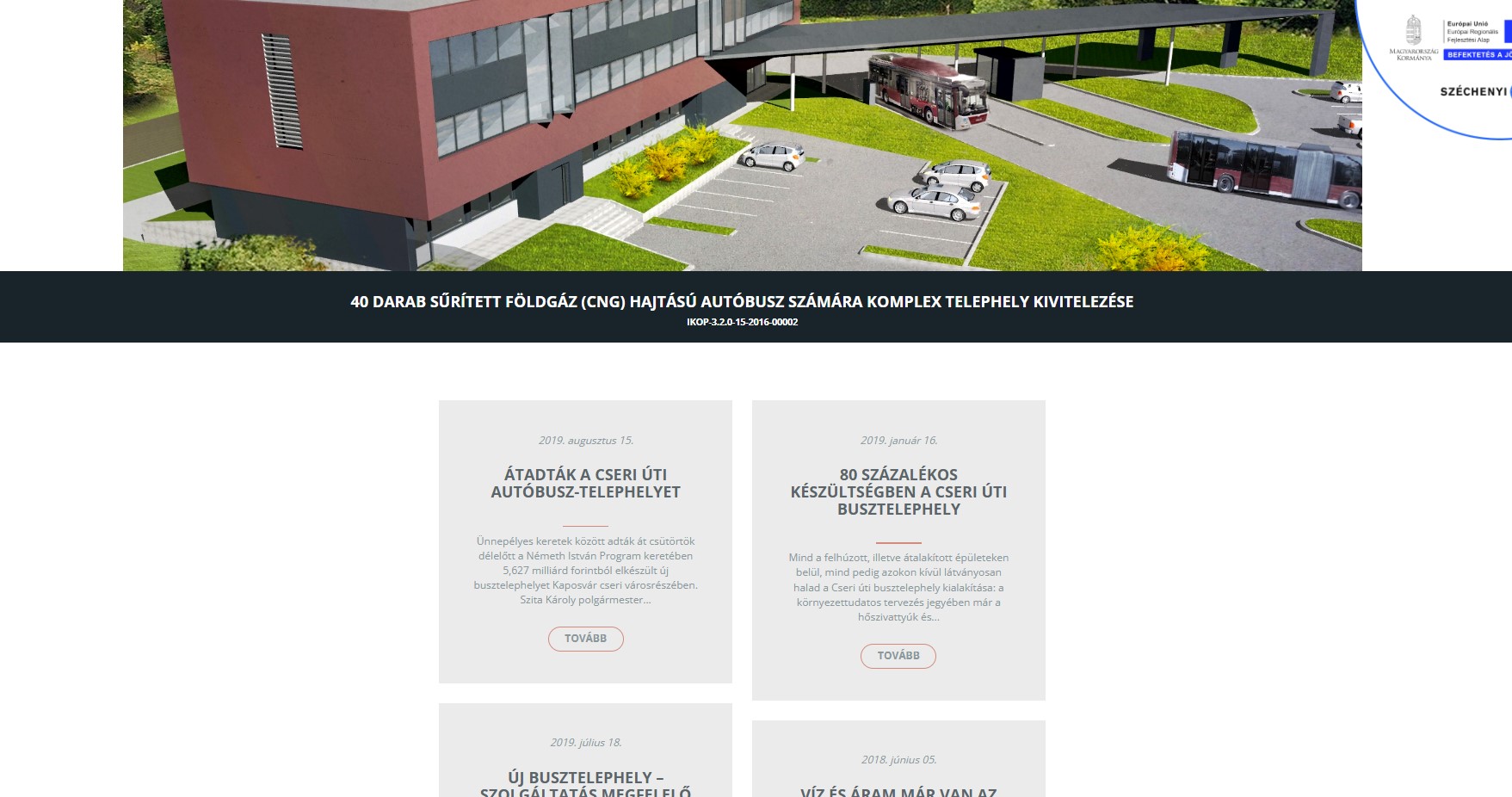
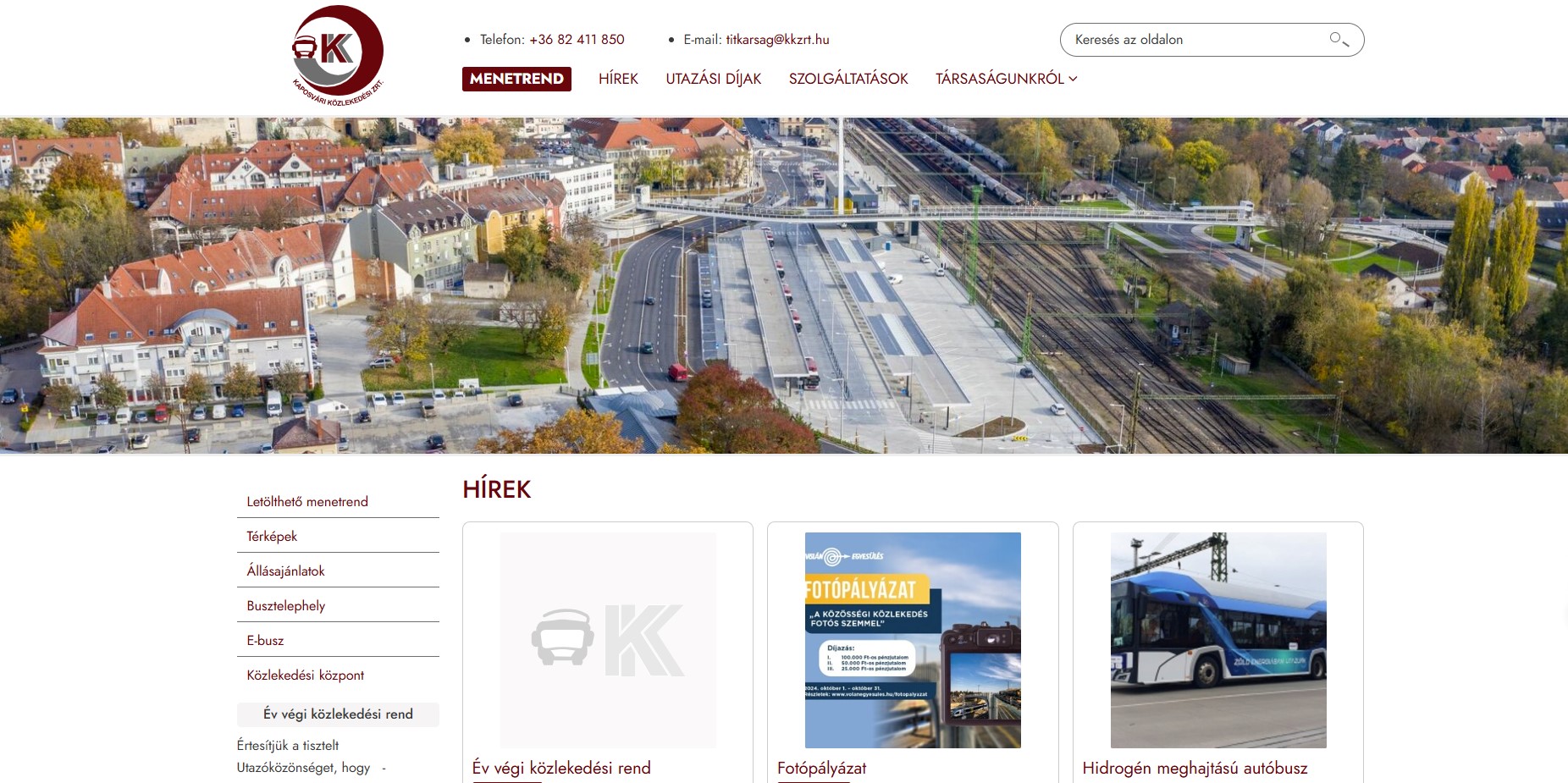
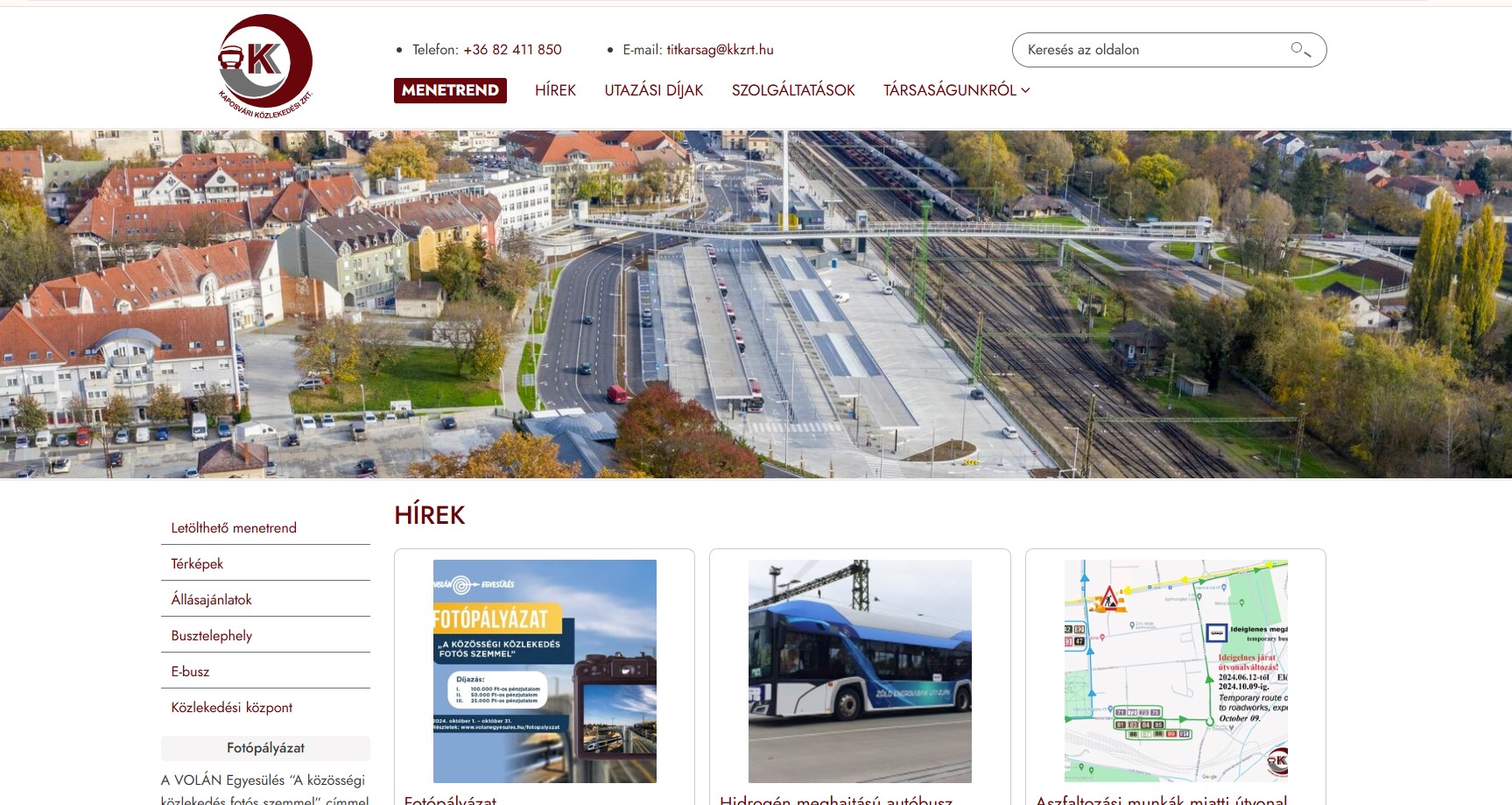
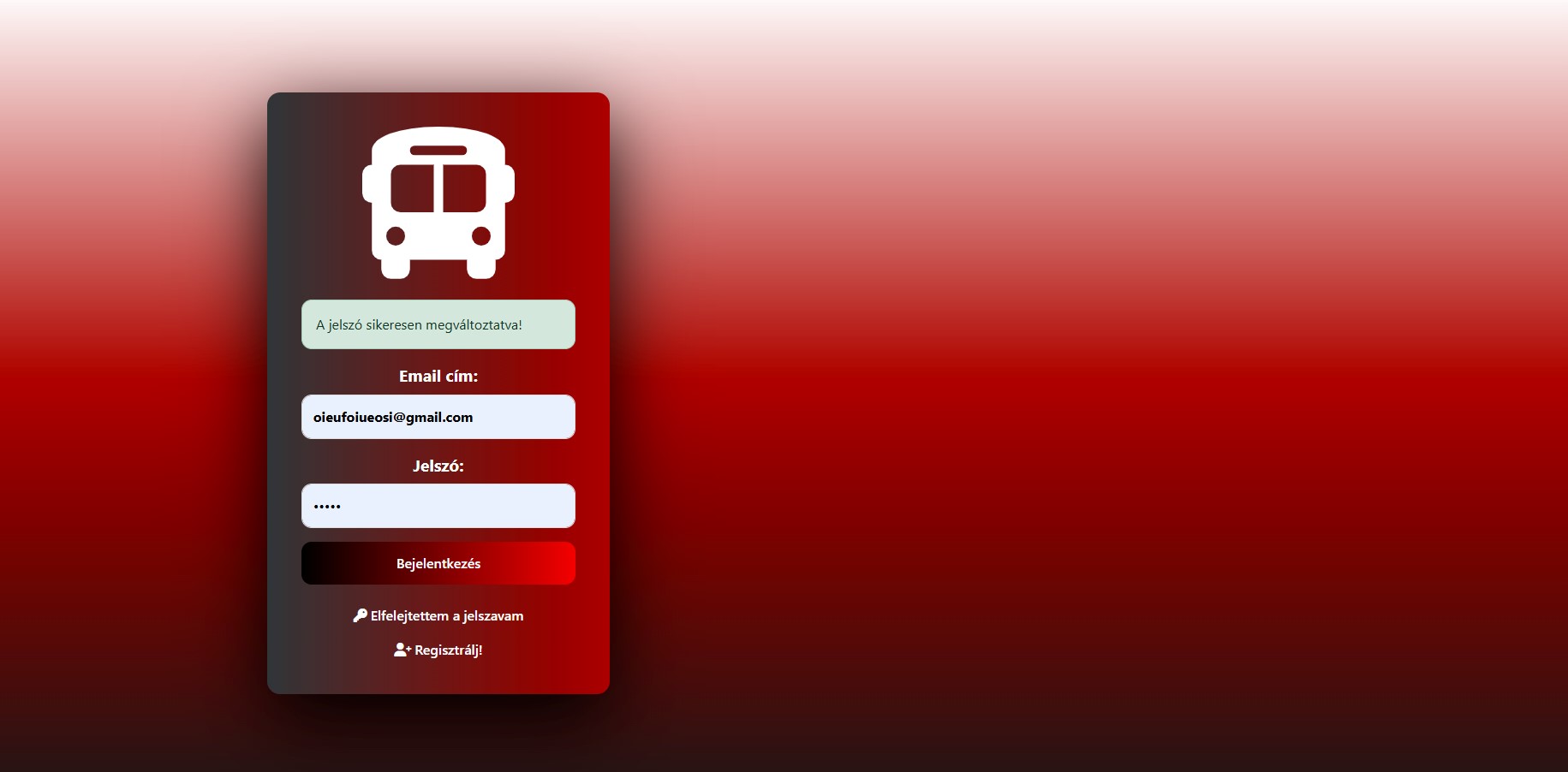
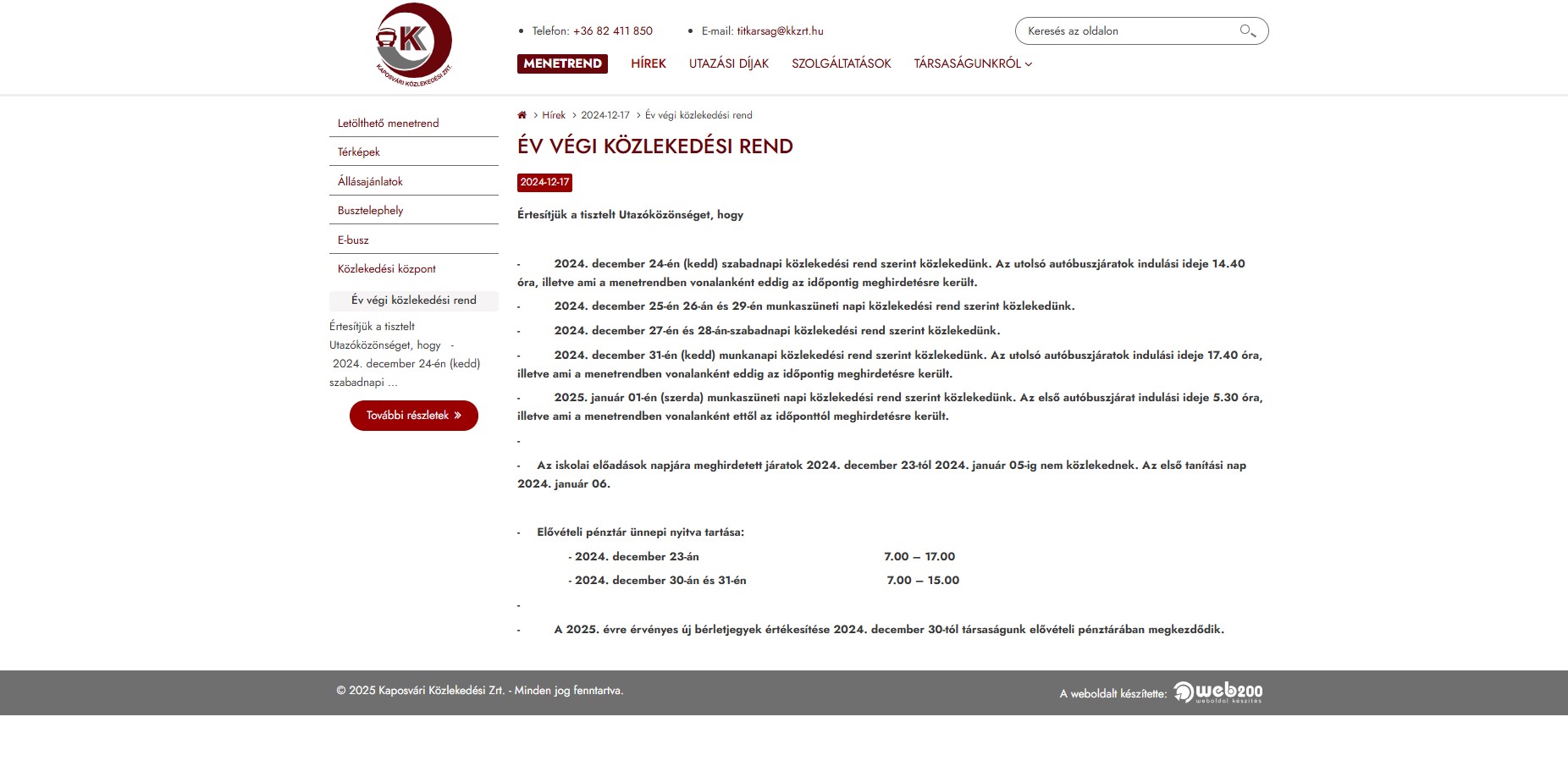
A felhasználói visszajelzések alapján elégedettlenséggel találkoztunk, saját magunk is tapasztaltuk a KKZRT weboldalának bonyolult felépítését.

Az oldal inspirációt ad egy modern közlekedési oldal létrehozásában.

A korábbi rendszer számos kritikus hiányossággal küzdött, amelyek megnehezítették az utazók mindennapi tapasztalatait. A papíralapú menetrendek, a statikus információs rendszerek.

felhasználói élményét javítása és a helyi járatos közlekedés fellendítése érdekében hoztuk létre a KaposTransit weboldalt.

<https://www.kaposbusz.hu/>Képek:



### 1.2 Stratégiai célkitűzések

A projekttel két fő dolgot szeretnénk elérni: jobbá tenni a városi közlekedést és korszerűsíteni azt.

A platform célja egy olyan felhasználóbarát rendszer létrehozása, amely valós idejű, naprakész információkkal szolgál az utazók számára, ez magában foglalja a dinamikus menetrendek, és az azonnali útvonaltervezési lehetőségek biztosítását, amely képes alkalmazkodni a felhasználók egyedi igényeihez.

A weboldal könnyen kezelhető és képes kiszolgálni mind a digitálisan képzett fiatal generációt, mind az idősebb korosztályt.

### 1.3 A Meglév**ő** rendszer korlátjainak átfogó elemzése

A korábbi közlekedésirányítási rendszer összetett technikai és felhasználói korlátokkal küzdött, amelyek jelentősen megnehezítették az utazók mindennapjait. A statikus PDF formátumú menetrendek nem csupán nehezen értelmezhetőek voltak, hanem azonnali frissítésük is rendkívül körülményes és időigényes folyamatnak bizonyult. Ez a modell nem vette figyelembe felhasználók azonnali információigényét.

A felhasználói felület hiányosságai tovább rontották a helyzetet. A nem reszponzív dizájn, a bonyolult navigáció és az elavult technológiai megoldások további akadályokat állítottak az utazók elé. Az információhiány pedig különösen kritikus problémát jelentett: a hiányzó valós idejű járatinformációk, a nem transzparens útvonaltervezési lehetőségek és a korlátozottan elérhető forgalmi adatok nagyban csökkentették a rendszer használhatóságát.

Ez az összetett kihívás tette szükségessé egy teljesen új, a kor technológiai színvonalának megfelelő digitális platform kifejlesztését, amely képes átfogó módon kezelni a helyi közösségi közlekedés valamennyi jelenlegi problémáját.

2. Rendszerkövetelmények:

### 2.1 Hardver követelmények

A KaposTransit tervezése során kiemelt figyelmet fordítottunk arra, hogy a rendszer a lehető legtöbb eszközön elérhető legyen. A hardver követelményeket úgy alakítottuk ki, hogy az elérhető legyen mind a legújabb, mind a kissé régebbi eszközökön.

A minimális hardver specifikáció gondosan lett meghatározva annak érdekében, hogy még a kevésbé korszerű eszközök is képesek legyenek zökkenőmentesen futtatni alkalmazásunkat. Az eszköztámogatás kiterjedt a hagyományos asztali számítógépektől kezdve a mobil eszközökön át egészen a táblagépekig, biztosítva ezzel a maximális hozzáférhetőséget.

### Eszköztípusok részletes támogatása

1. Asztali számítógépek és laptopok A platformot úgy terveztük, hogy zökkenőmentesen működjön a különböző hardverkonfigurációjú számítógépeken. A minimális elvárások magukban foglalnak egy alapszintű processzort, elegendő memóriát és grafikai képességeket a zökkenőmentes böngészéshez és térképes funkciókhoz.
2. Mobil eszközök Kiemelt figyelmet fordítottunk a mobil eszközök támogatására, mivel a felhasználók jelentős része okostelefonról vagy táblagépről éri majd el a szolgáltatást. Az alkalmazás reszponzív dizájnja biztosítja, hogy az összes funkció elérhető legyen a különböző képernyőméreteken.

#### 2.2 Szoftver követelmények

A KaposTransit fejlesztése során alapvető célunk volt, hogy a lehető legszélesebb körben biztosítsuk a kompatibilitást, figyelembe véve a felhasználók eltérő szoftverkörnyezetét.

### Böngésző támogatás

A platformot úgy terveztük, hogy kompatibilis legyen több böngészővel is, biztosítva a zökkenőmentes, biztonságos böngészést.

1. Asztali böngész**ő**k Google Chrome Böngészőben, valamint Microsoft Edge-ben teszteltük

### Speciális szoftver követelmények

A modern szoftverfejlesztés elengedhetetlen eleme a JavaScript-támogatás, amely biztosítja a webes alkalmazások dinamikus működését. A HTML5 szabvány támogatása jelentős a mai webes megoldásoknál. A korszerű adatbeviteli mezők implementálása tovább növeli a felhasználói élményt és a felület használhatóságát. Elengedhetetlen a folyamatos böngészőkompatibilitás fenntartása, rendszeres ellenőrzésekkel. Ez garantálja, hogy a szoftver rugalmasan alkalmazkodjon a változó technológiai környezethez és felhasználói igényekhez.

#### 2.3 Hálózati követelmények: folyamatos és biztonságos kapcsolat

A KaposTransit hálózati követelményrendszere úgy került kialakításra, hogy biztosítsa a folyamatos, biztonságos és hatékony adatkapcsolatot, minimális felhasználói beavatkozás mellett.

### Kapcsolat típusok és min**ő**ségi elvárások 1. Internetkapcsolat típusok

Vezetékes internetkapcsolat

WiFi hálózatok

Mobil adatkapcsolatok (4G/5G)

### 2. Sávszélesség elvárások

Letöltési sebesség: 5 Mbps

Feltöltési sebesség: 1 Mbps

Ajánlott sávszélesség: 20 Mbps

### Adatforgalmi jellemz**ő**k

Minimális havi adatforgalom: 50 MB

Ajánlott havi adatforgalom: 500 MB - 2 GB

Folyamatos adatszinkronizálás

Alacsony késleltetésű kapcsolat

### Biztonsági protokollok

Végponttól végpontig terjedő adatvédelem

Biztonságos hálózati kommunikáció Automatikus kapcsolatkezelés

### Offline üzemmód támogatása

A rendszer képes alapvető funkciók biztosítására korlátozott vagy nem elérhető internetkapcsolat esetén is:

Letöltött menetrendek megtekintése

Alapvető térképes funkciók használata

A hálózati követelményre vonatkozó részletes specifikáció biztosítja, hogy a KaposTransit szinte bármilyen körülmények között elérhető és használható legyen a felhasználók számára.

## 3. Első lépések

### 3.1 Regisztrációs folyamat: belépés

A felhasználói regisztráció az első és legfontosabb lépés a KaposTransit használatbavételekor. Gondosan megtervezett folyamatunk célja, hogy egyszerű, biztonságos és felhasználóbarát legyen.

### Regisztráció lépésr**ő**l lépésre

1. Platform elérése A regisztrációs folyamat első lépéseként a felhasználónak fel kell keresnie a KaposTransit weboldalát.



2.

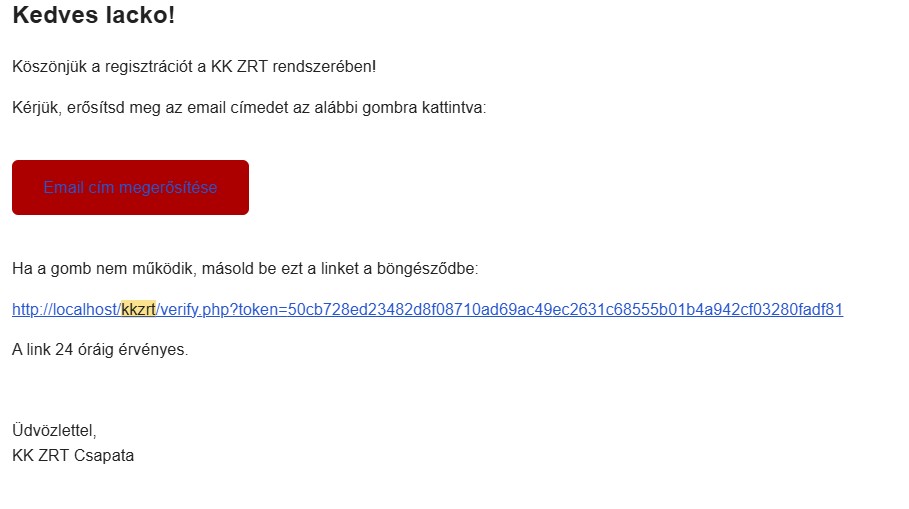
Regisztrációs űrlap kitöltése

A felhasználónak egy átfogó, de könnyen kitölthető űrlapot kell kitöltenie. A kötelező mezők gondosan lettek kiválasztva annak érdekében, hogy csak a legszükségesebb információkat kérjük el, szükséges felhasználónév, érvényes e-mail cím, valamint jelszó.

3.

Jelszó biztonság

A jelszó létrehozásakor kiemelt figyelmet fordítunk a biztonságos tárolásra.



5.

Meger**ő**sít**ő** folyamat

A regisztráció nem tekinthető véglegesnek az e-mail cím megerősítéséig, a weboldal regisztráció után aktiváló linet küld a megadott e-mail címre, 24 órán belül szükséges az aktiválás, sikertelen aktiválás esetén a regisztráció törlésre kerül.



Speciális regisztrációs feltételek

Egy e-mail cím csak egy fiókhoz használhatóm, valós személyes adatokat kell megadni

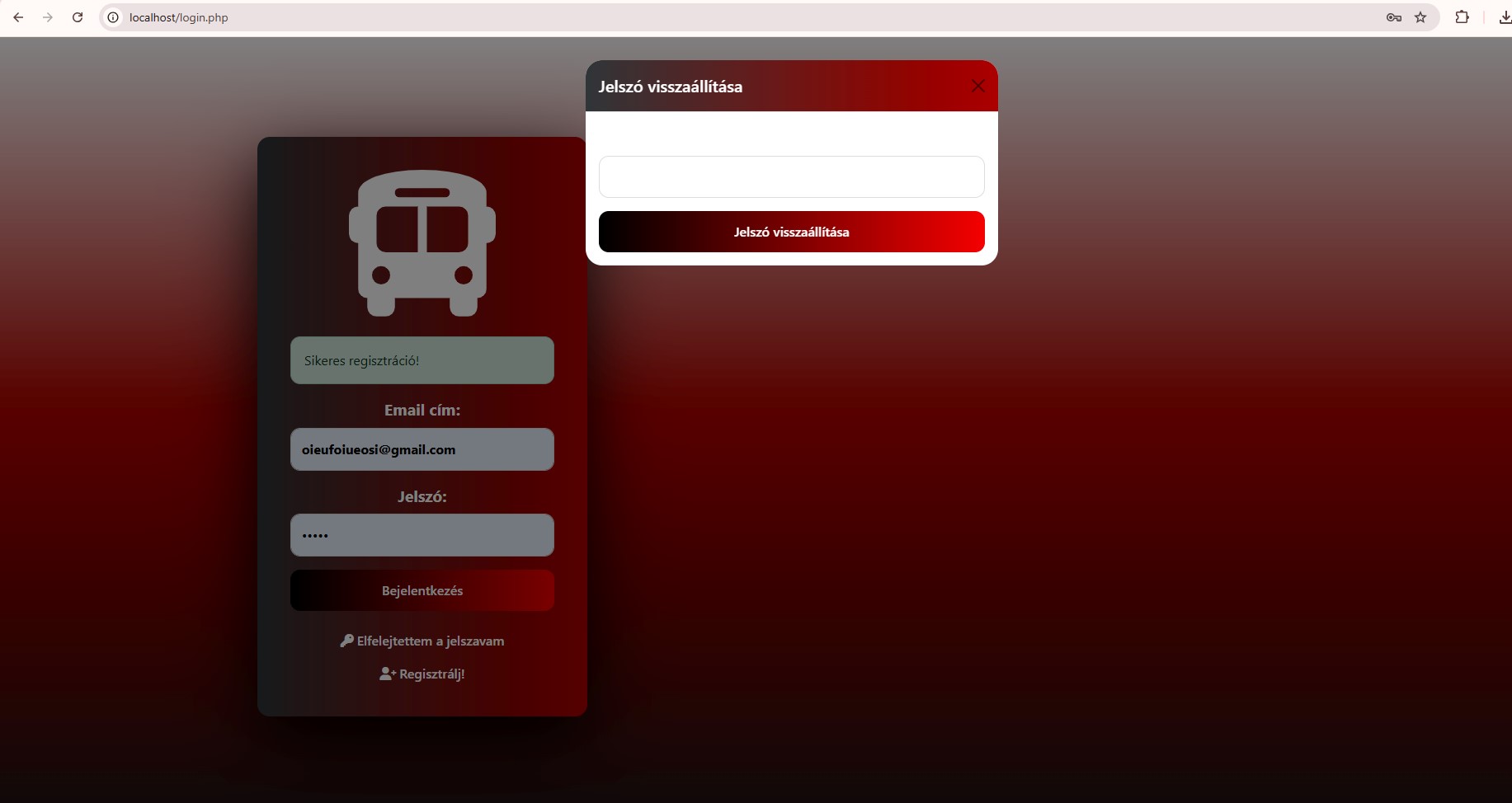
3.2 Bejelentkezési lehetőségek: biztonság és kényelem

Hagyományos bejelentkezés

* 1. Felhasználónév vagy e-mail Regisztrált e-mail cím, jelszó megadása, automatikus

hibajelzés érvénytelen hitelesítő adatok esetén,

* 1. Jelszó kezelés Elfelejtett jelszó helyreállítása, biztonságos jelszócsere, gyors és egyszerű regisztráció



Kétfaktoros azonosítás

Az e-mail címet phpmailer segítségével hitelesítjük megerősítés során.

## 4. Fő Funkciók: a digitális mobilitás összetett rendszere

### 4.1 Menetrendek: dinamikus információs rendszer

Egy olyan rendszert hoztunk létre, amely folyamatosan frissül és alkalmazkodik a pillanatnyi forgalmi viszonyokhoz.

Térképes megjelenítés A weboldal Kaposvár térség helyi járatos adatok feldolgozásával készült. A Valós idejű frissítések lehetővé teszik, hogy a felhasználó meg tudja tekinteni a következő induló járatot. Részletes járatinformációk az következő induló járat megállóiról. Keresőmező segítségével járatszám szerinti rendezést, azok hétvége és hétköznapi szétválogatását oldottuk meg. Mind ehhez interaktív várostérkép is tartozik, amelyen a megállókat koordinátákkal-jelölőkkel ábrázoltuk.

Keresési lehetőségek A térképen legördülő listából választhatja ki a felhasználó a járatot.

### 4.2 Célállomás kiválasztása

A projekt létrehozása során kiemelt figyelmet fordítottunk a földrajzi pontosságra, mind ehhez Google maps API-t használtunk, a kezdőponttól végpontig való keresés, a megállók koordinátái alapján működő megálló kereső opció ennek segítségével történik, a megállók nevei API alapján dinamikusan koordinátából történik.

### 4.3 Járatállapot figyel**ő**: valós idej**ű** mobilitási információk

#### 4.3.1 Valós idej**ű** járatinformációk

1. Pontos érkezési adatok Szintén a Google maps API segítségével elértük azt, hogy a MÁV, valamint a Volán által nyilvántartott indulásidők pontosan hozzá legyenek rendelve a keresés opcióhoz, eredményképp pontos indulásidőt és valós idejű érkezést kaptunk vissza.



## 5. GTFS integráció

### 5.1 GTFS formátum áttekintés

A GTFS (General Transit Feed Specification) egy nemzetközileg elismert adatformátum, amely forradalmasította a közösségi közlekedési információk megosztását és felhasználását. Ez a szabványosított formátum lehetővé teszi, hogy a közlekedési szolgáltatók egységes módon tegyék elérhetővé menetrendi adataikat, útvonalaikat és állomásaik információit.

Míg a múltban minden helyi közlekedési vállalat egyedi, zárt rendszereket fejlesztett online menetrendi felületeikhez, addig napjainkra a GTFS vált az iparági normává. Ez a nyílt standard nem csak a fejlesztési időt és költségeket csökkenti, hanem jelentősen jobb felhasználói élményt is biztosít.

A GTFS adatbázis lényegében több, egymással összefüggő CSV fájlból áll, amelyek tartalmazzák a járatok, megállók, menetrendek és egyéb fontos közlekedési információk strukturált gyűjteményét. A rendszer legnagyobb előnye, hogy dinamikus és rugalmas - képes kezelni a különleges forgalmi helyzeteket, járatkimaradásokat vagy útvonal-módosításokat is.

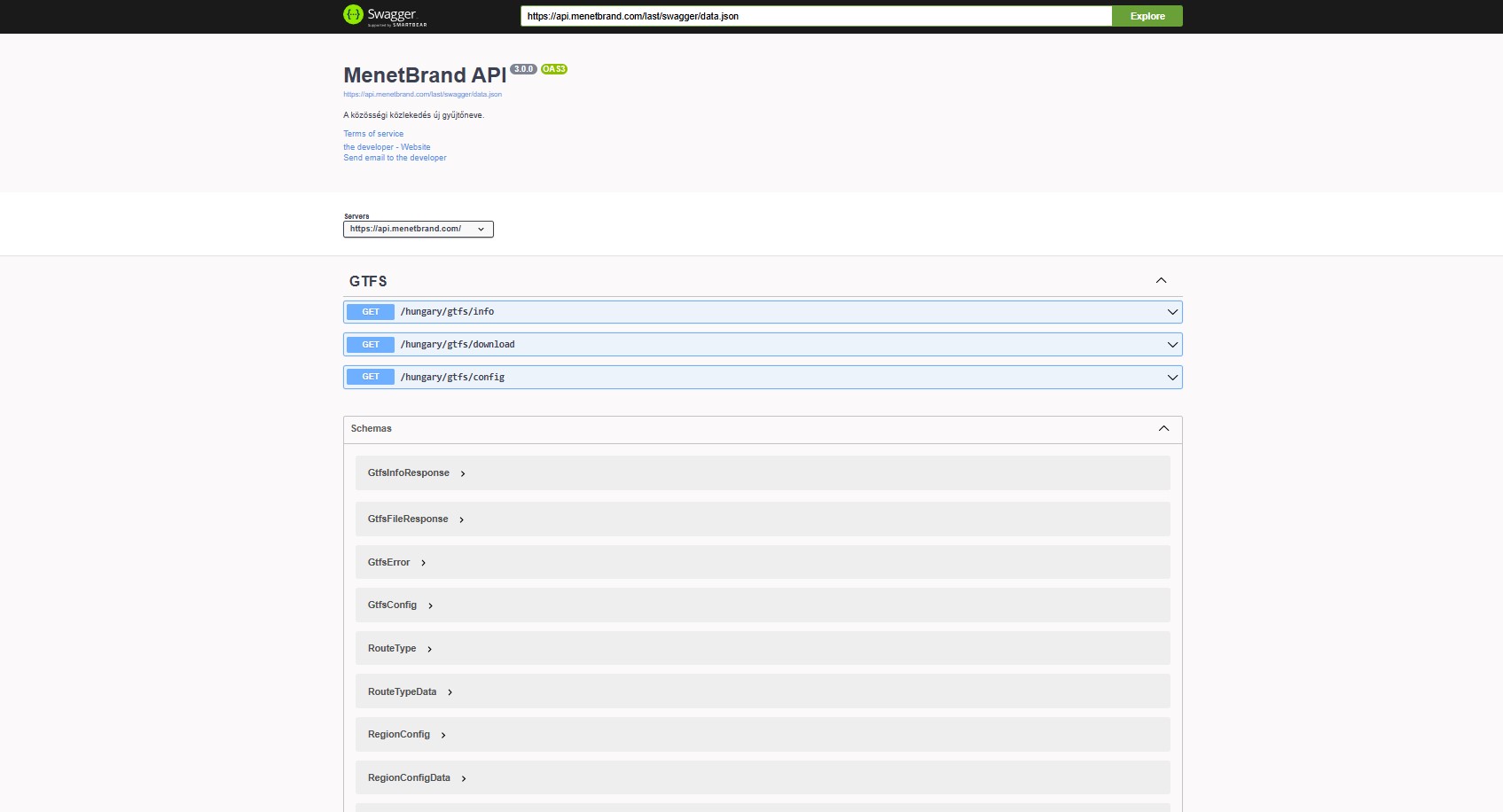
A KaposTransit rendszerünkben ezt a nemzetközi szabványt vettük alapul, de számos területen továbbfejlesztettük azt. Kiegészítő számításokat, optimalizációs algoritmusokat és bővített adatstruktúrát alkalmazunk, hogy a lehető legpontosabb és leghasználhatóbb információkat biztosíthassuk felhasználóink számára.

### 5.2 Támogatott partnerek

Rendszerünk alapját a Menetbrand magyar GTFS szolgáltató adja, akitől egyedi, kifejezetten a kaposvári térségre optimalizált GTFS adatbázist kaptunk. Ez az együttműködés lehetővé teszi számunkra, hogy a legfrissebb és legpontosabb információkkal szolgálhassunk felhasználóinknak.

A kapott adatbázis a Kaposvár központú, 10 kilométeres sugárú körben található összes közlekedési információt tartalmazza. Minden olyan járat, amely ezen a területen közlekedik vagy érinti azt, szerepel az adatbázisunkban. Ez magában foglalja nem csak a városi, hanem a helyközi járatokat is, biztosítva a teljes körű tájékoztatást az utazók számára.

Az adatbázisunk minőségét a rendszeres frissítések biztosítják: partnereink naponta, jellemzően reggel 6 és 9 óra között készítik el és teszik elérhetővé az aktualizált adatállományt. Ez garantálja, hogy alkalmazásunk mindig a legfrissebb menetrendi információkat közvetítse felhasználóink felé.



## 6. API dokumentáció

### 6.1 API áttekintés

A KaposTransit rendszer háttérben működő technológiánkat úgy terveztük, hogy egyszerre legyen modern, megbízható és könnyen bővíthető. Adataink két jól szervezett MySQL adatbázisban találhatók: a "kaposvar" adatbázis a város közlekedési alapinformációit, míg a "kkzrt" adatbázis a felhasználók adatait tartalmazza.

Hogy a felhasználók számára zökkenőmentes élményt biztosítsunk, szerverünk Node.js technológiára épül, amely lehetővé teszi a gyors válaszidőket és a megbízható működést még csúcsidőben is. A rendszer a 3000-es porton fut, ahol az adatokat rendezett, könnyen feldolgozható JSON formátumban szolgáltatjuk ki.

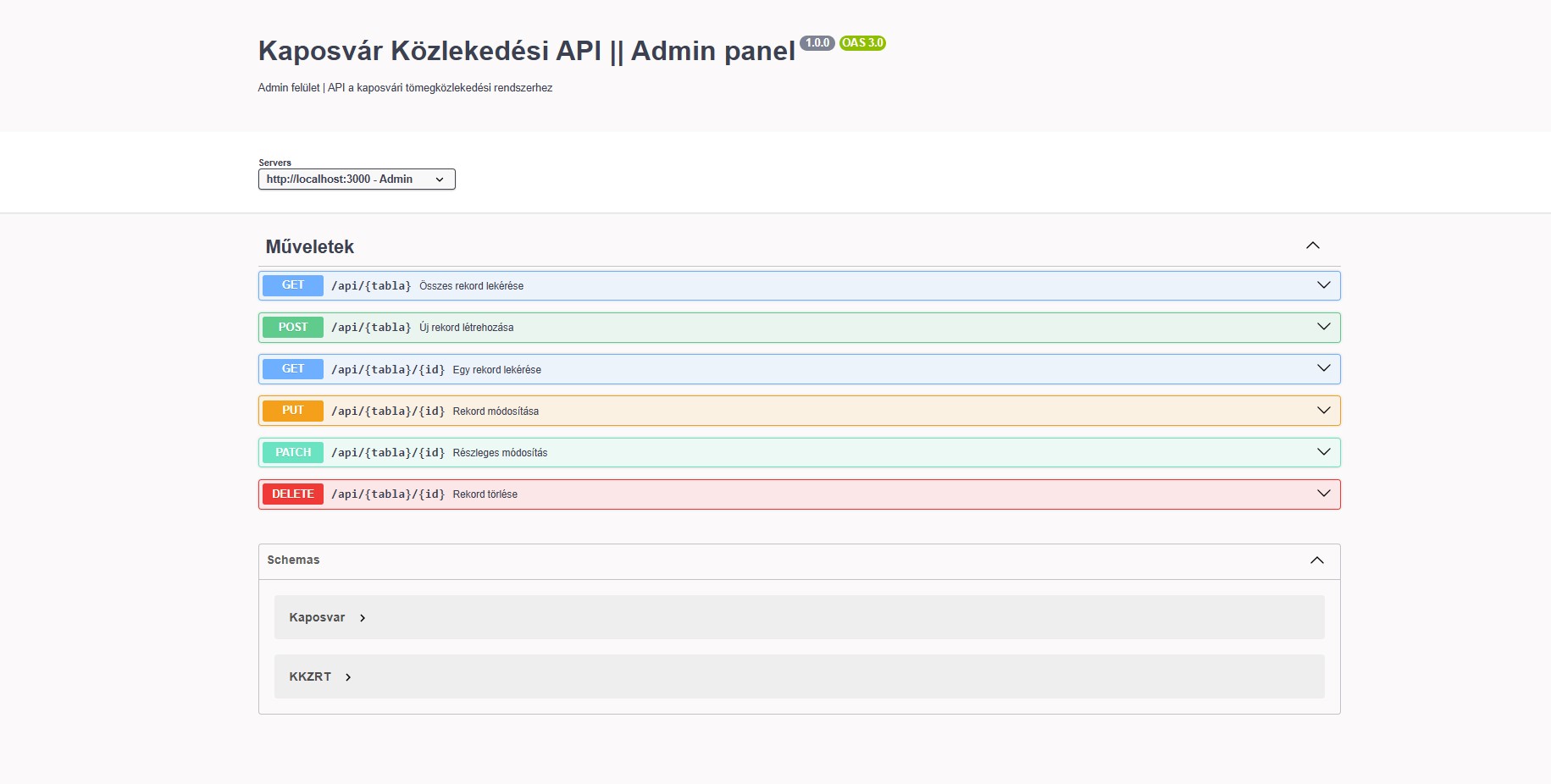
Fejlesztőink számára kiterjedt dokumentációt biztosítunk Swagger felületen keresztül, amely interaktív módon mutatja be az elérhető API végpontokat. Ez a felület nem csak az adatok lekérdezését, hanem a tesztelést is egyszerűvé teszi. Munkánk során mi magunk is aktívan használjuk a Postman alkalmazást a végpontok tesztelésére és finomhangolására.

Az API-n keresztül elérhető adatok egyszerű URL-eken keresztül érhetők el, így akár külső rendszerek is könnyen integrálhatók a KaposTransit szolgáltatásaival. Az adatok módosítása és új elemek hozzáadása szintén dokumentált folyamatokon keresztül történik, biztosítva a rendszer stabilitását és biztonságát.

### 6.2 Végpontok

#### 6.2.1 Menetrend végpontok

A KaposTransit rendszer egyik legfontosabb funkcióját a menetrendi adatokat szolgáltató API végpontok biztosítják. Ezeken keresztül érhető el minden információ, amely a járatok közlekedéséhez kapcsolódik.



\*\*Főbb menetrendi végpontok: \*\*

/api/stop: Az összes elérhető MÁV megálló

/api/route: Az összes elérhető MÁV útvonal

/api/marker: Az összes kaposvári helyi járatos autóbusz neve, koordinátái

/api/buszjarat: Az összes kaposvári helyi autóbusz útvonalai

Ezek a végpontok lehetővé teszik, hogy az alkalmazásunk mindig naprakész menetrendi információkkal szolgáljon a felhasználók számára. A válaszok tartalmazzák az időpontokat, útvonalakat, megállókat és minden olyan adatot, amely egy utazás megtervezéséhez szükséges.

F**ő**bb járatállapot végpontok:

Ezek a végpontok biztosítják, hogy a felhasználók mindig naprakész információkkal rendelkezzenek a járatok tényleges közlekedéséről, így hatékonyan tervezhetik útjaikat és elkerülhetik a váratlan késéseket.

### 6.3 Példa kérések és válaszok

Az API használatának megkönnyítése érdekében néhány mintát mutatunk be a leggyakoribb kérésekre és az azokra adott válaszokra.

Példa megálló adatainak lekérdezésére:

Kérés:

**text** GET /api/marker

Válasz:

**JavaScript**

{

"nev": "Pázmány Péter utca",

"lat": 46.3651,

"lng": 17.799

},

{

"nev": "Pázmány Péter utca",

"lat": 46.3652,

"lng": 17.7984

},

{

"nev": "Klebelsberg Kollégium",

"lat": 46.3655,

"lng": 17.8074

},

{

"nev": "ÁNTSZ",

"lat": 46.365, "lng": 17.7893

},

Kérés:

**text** GET /api/buszjaratok

Válasz:

**JavaScript**

{

"route\_id": 1,

"number": "12",

"name": "Helyi autóbusz-állomás - Sopron u. - Laktanya", "name\_back": "Laktanya - Sopron u. - Helyi autóbuszállomás",

"goes\_back": 1,

"schedule\_id": 1,

"direction": "forward",

"stop\_id": 1,

"stop\_name": "Helyi autóbusz-állomás",

"stop\_time": "05:00:00",

"weekend": null,

"weekday": "weekday"

},

{

"route\_id": 2,

"number": "12",

"name": "Helyi autóbusz-állomás - Sopron u. - Laktanya", "name\_back": "Laktanya - Sopron u. - Helyi autóbuszállomás",

"goes\_back": 1,

"schedule\_id": 1,

"direction": "forward",

"stop\_id": 2,

"stop\_name": "Corso",

"stop\_time": "05:01:00",

"weekend": null,

"weekday": "weekday"

},

Ezek a példák jól szemléltetik, hogyan lehet a KaposTransit API-t használni a közlekedési információk lekérdezésére.

## 7. Mobil kompatibilitás: mobilitás a mobilon

### 7.1 Reszponzív dizájn: zökkenőmentes felhasználói élmény minden eszközön

A KaposTransit mobil kompatibilitása nem csupán egy technikai megoldás, hanem egy átfogó felhasználói élményt biztosító stratégia. A reszponzív dizájn alapelveit maximálisan érvényesítve hoztuk létre platformunkat, amely képes alkalmazkodni bármilyen eszköz képernyőméretéhez és felbontásához.

### Dizájn alapelvek

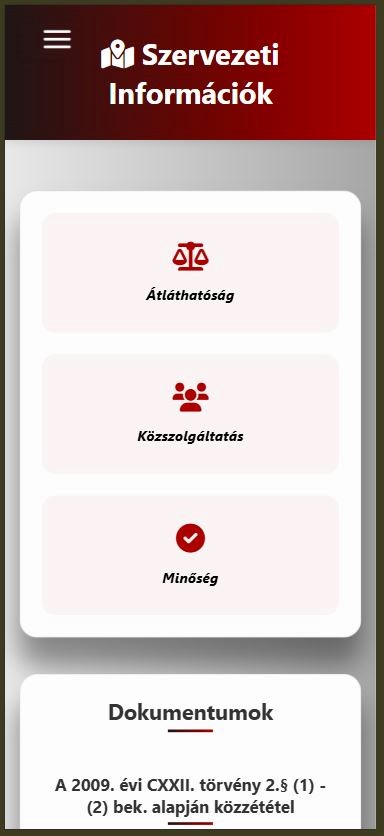
1. Adaptív felületarchitektúra A létrehozás során külön figyelmet fordítottunk arra, hogy eszközfüggetlen felhasználói felületet hozzunk létre
2. Érint**ő**képerny**ő**s optimalizáció A tesztelés során mobilos nézetben nagy érintési felületek, valamint ujjbarát navigációs elemek, támogatja az interakciókat.
3. Teljesítmény-optimalizálás Különös figyelmet fordítottunk a gyors betöltési időkre, ehhez próbáltuk optimalizálni az időigényes feladatokat, mindemellett hatékony erőforrás-kezelést is akartunk alacsony energiafogyasztással, ami részben megvalósult.

### Támogatott eszköztípusok

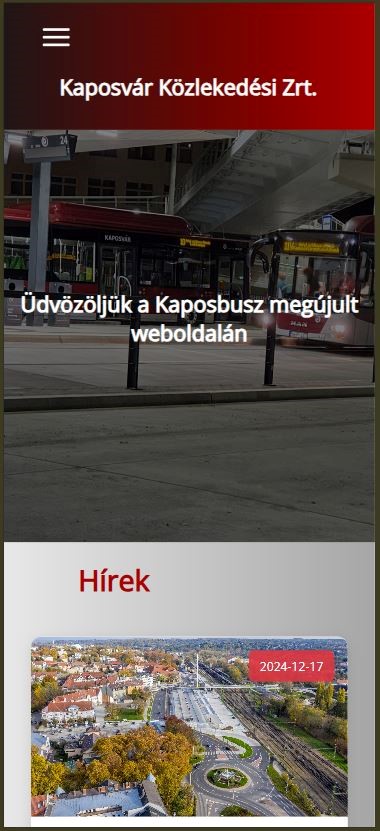
Projektünket okostelefonokon is teszteltük Android valamint IOS operációs rendszerrel egyaránt.



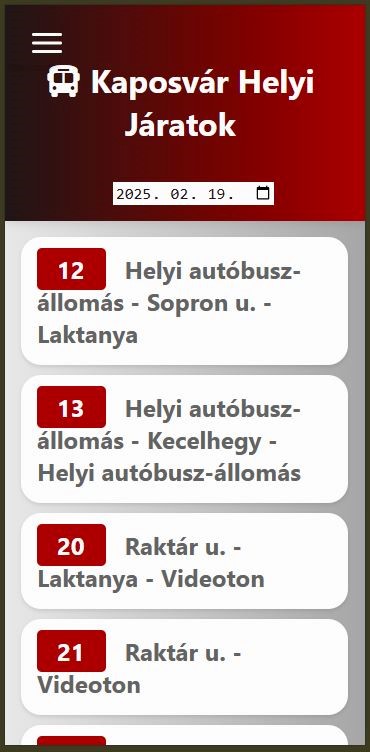
### Samsung Z Fold 5 Menetrend oldalon



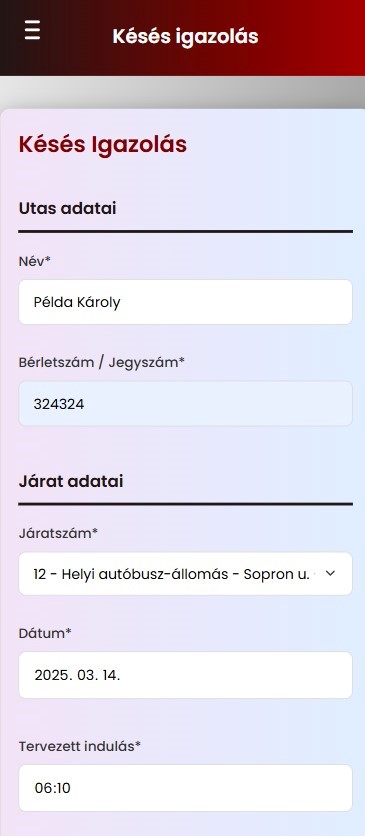
### Samsung Galaxy S20 Ultra Info oldalon



### Pixel 7 F**ő**oldalon

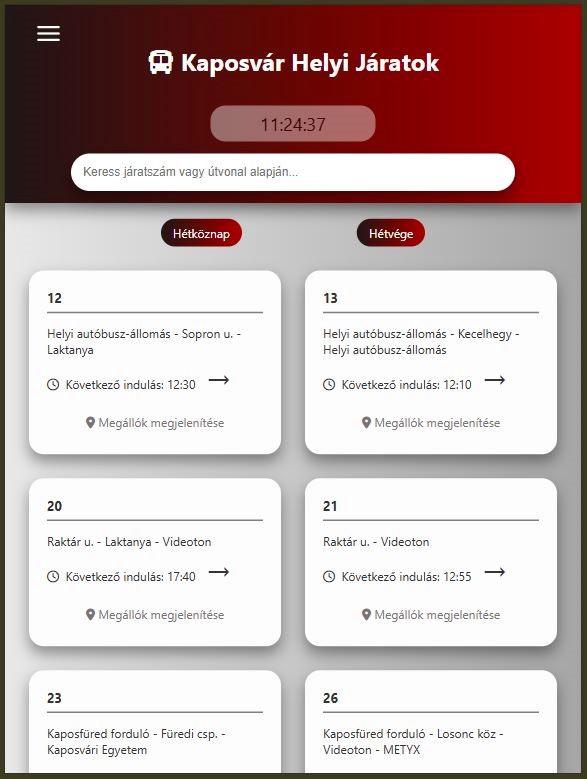


### Samsung Galaxy S8+ Járatok oldalon



IPhone XR Késés Igazolás oldalon

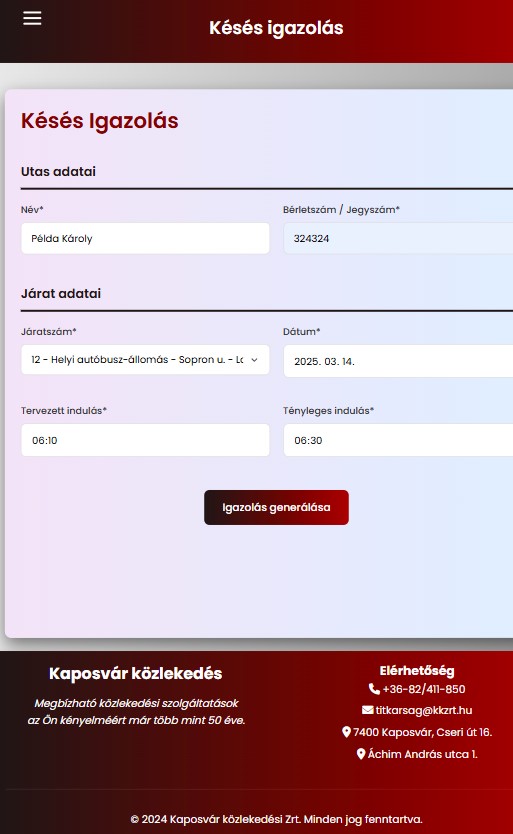
- Táblagépek



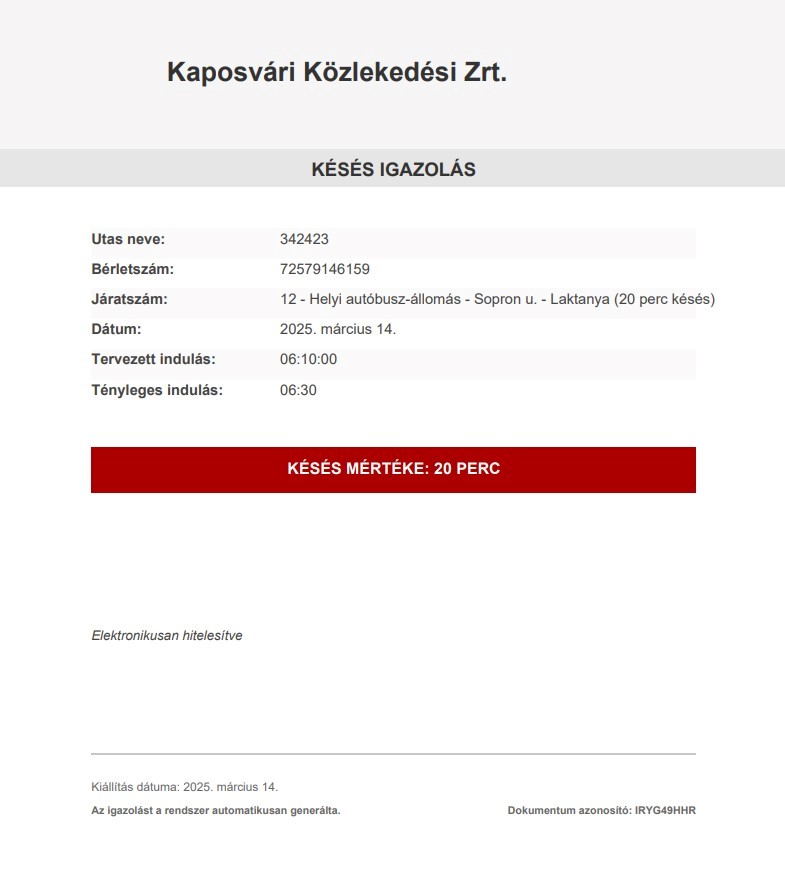
### IPad Mini Menetrend oldalon



### IPad Pro Járatok oldalon



### Asus Zenbook Fold Késés igazolás oldalon



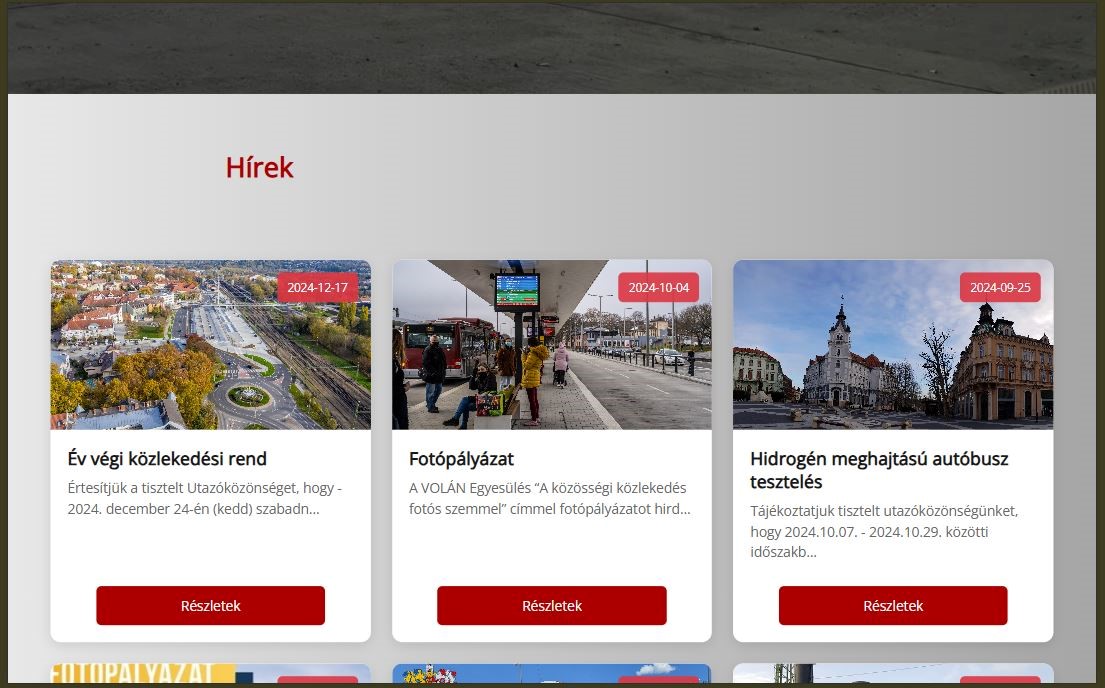
Kisméret

ű

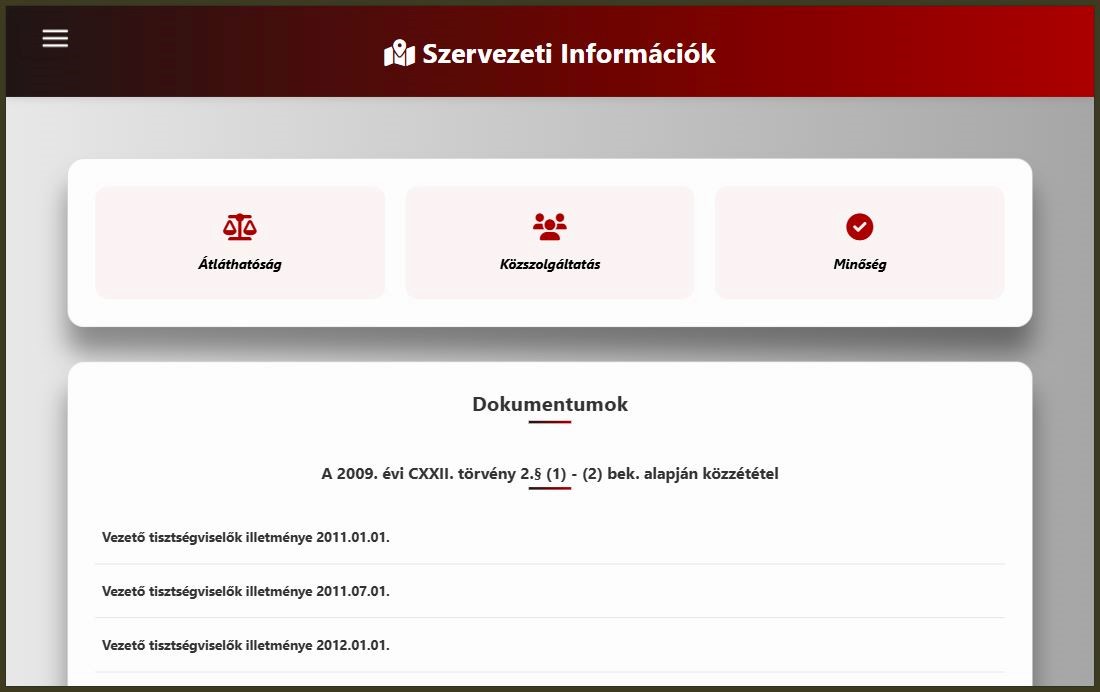
és nagy képerny

ő

s eszközök



Nest Hub Max F**ő**oldalon híreknél



Nest Hub Max Info oldalon

Érintőképernyős és hagyományos eszközök

## 8. Biztonsági és adatvédelmi szabályzat: átláthatóság és biztonság

### 8.1 Felhasználói adatok védelme: bizalmas adatkezelés

A KaposTransit elsődleges célja, hogy a legmagasabb szintű adatvédelmi és biztonsági standardokat alkalmazza, biztosítva felhasználóink személyes adatainak maximális védelmét.

### Adatvédelmi alapelvek

1. Adatminimalizálás Kizárólag a szükséges adatokat gyűjtünk, amelyek a későbbiek során jegyvásárlás opcióhoz, valamint felhasználói fiók használatához alkalmazunk. Mindez célhoz kötött adatkezelés.
2. Bizalmasság Projekt létrehozása során törekedtünk magas körű adatvédelemre, de a jövőben sokkal erősebb védelmet alakítunk ki.

### Kezelt adattípusok

Személyes azonosító adatok Jelenleg Felhasználó nevet, valamint e-mail címet és jelszót tárolunk, amely majd a későbbiek során a jegyvásárlás valamint a felhasználói fiók opcióhoz rendelünk.

8.2 Adatkezelési tájékoztató: jogok és kötelezettségek

### Adatkezelés részletei

1. Adattárolási időtartamok A Személyes adatokat folyamatos tároljuk, nem titkosított, de máshol tároljuk adatokat.
2. Adatok felhasználásának célja a jogszabályi kötelezettségeknek eleget tesz.

9. Tesztelés:

#### 9.1 Automatizált tesztelés

Az automatizált tesztelés Playwright-al lett letesztelve, ami egy gyors és megbízható end-to-end tesztelést biztosít modern weboldalakra/webalkalmazásokra.

Az automatizált tesztelésnél ellenőriztük, hogy minden oldal rendesen betöltött-e.

**JavaScript**

for (const pageInfo of pages) { test(`${pageInfo.title} oldal betöltése`, async ({ page }) =>

{ await page.goto(`http://localhost/kkzrt/${pageInfo.url}`); await

expect(page).toHaveURL(`http://localhost/kkzrt/${pageInfo.url}`);

});

}

A program végig megy a pages listán ahol el van tárolva az egyes oldalak url-je, amit úgy ellenőriztünk, hogy a tesztelő programnak megadott url megegyezik-e a kapott url-lel.

Ezentúl ellenőrzi, hogy az egyes API végpontok lekérései igaz értékkel tértek vissza.

**JavaScript**

for (const endpoint of workingApiEndpoints) { test(`${endpoint.name} API végpont ellenőrzése`, async ({ request }) => { const response = await request.get(`http://localhost:3000/${endpoint.url}`); expect(response.ok()).toBeTruthy();

});

}

Ugyan úgy, mint az előbb van egy listánk, ami végig megy a programrészen és lekérdezéseket tesz, várva, hogy true értékkel térjen vissza, hogy sikeresnek ítélje a program a tesztesetet.

Ezek után jönnek a komponensek tesztelése, mi megnézi, hogy a térképek láthatóra van-e állítva,

**JavaScript**

test('Térkép megjelenítés', async ({ page }) => { await page.goto('http://localhost/kkzrt/terkep.php'); await expect(page.locator('#map')).toBeVisible();

await page.goto('http://localhost/kkzrt/megallok\_kereso.php'); await expect(page.locator('#map')).toBeVisible();

});

mindezt azzal, hogy megkeresi a térképeket és, hogy azok értéke láthatóra van-e állítva. (Érthetően ez a teszteset nem ellenőrzi, hogy helyesen működik-e a térképek, ami a mi esetünkben nem menne át a teszten az Google Maps API kulcs hiánya miatt.) A következő komponens, amit tesztel, azaz összes hírnek a kilistázása.

**JavaScript**

test('Az összes hír kilistázása', async ({ page }) => { await page.goto('http://localhost/kkzrt/index.php'); const btn = await page.locator('#btnMoreNews');

let isExpanded = await page.evaluate(() => window.isExpanded = false);

expect(isExpanded).toBeFalsy(); await btn.click();

isExpanded = await page.evaluate(() => window.isExpanded = true); expect(isExpanded).toBeTruthy();

});

A teszteset megnézi, hogy a gomb megnyomása után az isExpanded értéke megváltozik-e igazra, ami a helyes működésnél is megváltozna.

Az utolsó komponens teszteset a Részletek gomb helyes átirányítását, teszteli.

**JavaScript**

test('Főoldalon részletek gomb ellenőrzése', async ({ page }) =>

{ await page.goto('http://localhost/kkzrt/index.php'); await page.click('text=Részletek'); await

expect(page).toHaveURL('http://localhost/kkzrt/news.php?id=1') });

A teszt közben elmegy a főoldalra, megnyomja a Részletek gombot és megnézi, hogy az adott oldalra vezetett-e át.

Az utolsó előtti része a programnak az API végpontok válaszát vizsgálják.

**javascript**

test('API válaszok ellenőrzése', async ({ request }) => {

*// Megállók API teszt*  const stopsResponse = await request.get('http://localhost:3000/api/stop'); const stopsData = await stopsResponse.json(); expect(Array.isArray(stopsData)).toBeTruthy();

*// A többi másik API végpontot is ugyanígy lett tesztelve* });

Ennél is lekérdezéseket végzünk, aminek a válaszának adatát eltároljuk és a program megnézi, hogy tömb-e a kapott adat.

Az utolsó teszteset megnézi, hogy az infó oldalon lévő linkek valós helyre mutatnak.

**JavaScript**

test('404-es linkek keresése', async ({ page }) => { await page.goto('http://localhost/kkzrt/info.php');

const links = await page.locator('a').all(); let brokenLinks = [];

for (const link of links) { const href = await link.getAttribute('href');

if (href && !href.startsWith('#')) {

const response = await page.request.get(href);

if (response.status() === 404) { console.log(`❌ 404 ERROR: ${href}`); brokenLinks.push(href);

}

} }

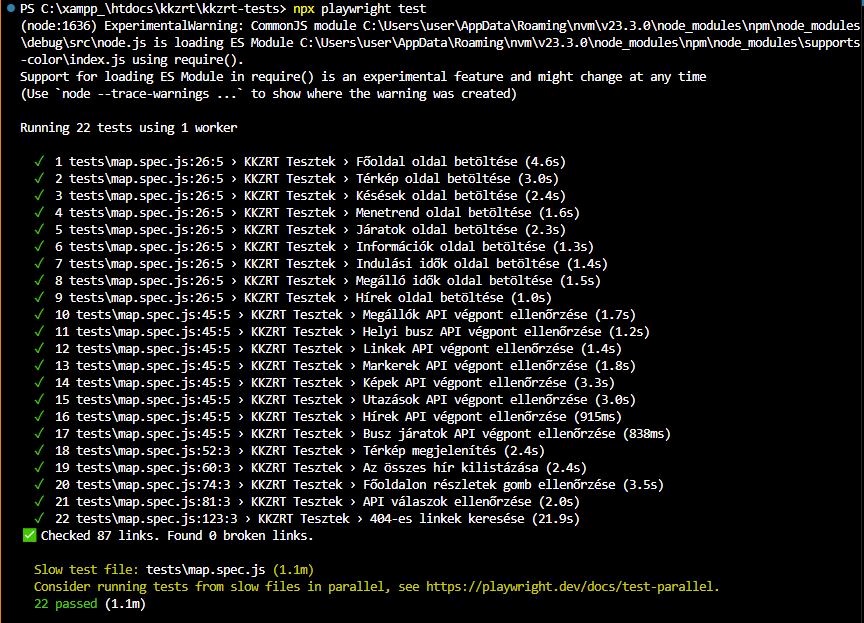
console.log(`✅ Checked ${links.length} links. Found

${brokenLinks.length} broken links.`);

*// Force the test to fail if any broken links exist*  expect(brokenLinks.length).toBe(0);

});

A program megnyitja az összes linket és, ha valamelyik 404-es oldalra mutat, akkor az adott linket belerakja egy listába és később kiírja. A teszt akkor sikeres, ha egyetlen link sem vezet 404-es oldalra.



Automatizált tesztesetek 9.2 Manuális tesztelés:



Manuális tesztesetek

További részletben kérjük nézze meg a mellékelt tesztesetek.xlsx fájlt, ahol le van írva az összes teszteset, amit manuálisan végeztünk.

## 10. Hibakeresés és támogatás

### 10.1 Gyakori hibajelenségek: diagnosztikai és megoldási útmutató

A KaposTransit célja, hogy a lehető legzökkenőmentesebb felhasználói élményt biztosítsa, ugyanakkor felkészültünk a felmerülő technikai kihívásokra is.

### Technikai hibák kategorizálása

#### 1. Bejelentkezési problémák

Elfelejtett jelszó

Egyéb hitelesítési nehézségek (E-mail cím) Javasolt megoldások:

Jelszó-visszaállítási folyamat

Ügyfélszolgálati segítségkérés

#### 2. Térképes funkciók hibái

Helymeghatározási pontatlanságok

Útvonaltervezési hibák

Térképes adatok elavultsága Javasolt megoldások:

Manuális helymegadás

Térképes adatok frissítése

Alternatív útvonalak keresése

10.2 Ügyfélszolgálati elérhet**ő**ségek: komplex támogatási rendszer

### Támogatási csatornák

#### 1. Telefonos ügyfélszolgálat

Közvetlen emberi segítségnyújtás

Munkanapokon: 4:15-22:30

Dedikált technikai támogatási vonal

Magyar nyelvű, képzett ügyintézők

#### 2. E-mail támogatás

Részletes problémaleírás küldése

Hivatalos támogatási e-mail cím

Válaszadási idő: 24-48 óra

Részletes dokumentáció csatolásának lehetősége

A hibakeresés és támogatás nem csupán egy technikai folyamat, hanem a KaposTransit azon elkötelezettsége, hogy minden felhasználónk zökkenőmentes és élménydús utazást tudjon tervezni.

## 11. Fejlesztési tervek

### 11.1 Jöv**ő**beni funkciók: technológiai horizont és innovációs stratégia

A KaposTransit fejlesztési stratégiája nem csupán a jelenlegi igények kiszolgálásáról szól, hanem egy folyamatosan megújuló, előremutató mobilitási ökoszisztéma létrehozásáról. Fejlesztési terveink négy fő pillérre épülnek: technológiai innováció, felhasználói élmény, rendszerintegráció és fenntarthatóság.

### Technológiai fejlesztések

1. Mesterséges intelligencia alapú megoldások A jövőben tervezzük az útvonaltervezés megújítását modernizálását, gépi tanuláson alapuló forgalmi előrejelzés beépítését, személyre szabott utazási ajánlatokkal bővítenénk. Valós idejű forgalomoptimalizálást szeretnék, valamint intelligens közlekedésirányítási rendszert kiépíteni.
2. Kiterjesztett valóság (AR) navigáció Valós idejű navigációs információkkal, megállóhelyek és útvonalak, valós környezetbe ágyazott utazási információkkal, interaktív térképes megoldásokkal, Látássérültek számára speciális navigációs segédeszközök hozzáadásával szeretnénk bővíteni projektünket ez mind támogatás függvényében.
3. Gépi tanulás és adatelemzés A gépi tanulás során szeretnénk, hogy mesterséges intelligencia alapján forgalmi mintákat elemezzen, értelmezzen, és létrehozzunk egy olyan rendszert, amely szinte az összes megyei jogú város helyi járatos közlekedés modelljének befogadására alkalmas. A használatával környezeti hatásokat, és olyan tényezőket is megfigyelhetünk, amelyek felhasználásával egy olyan rendszert tudnánk kiépíteni, amely megállja a helyét és képes felvenni a versenyt. (AI alapú rendszerekkel)

### Felhasználói élmény fejlesztése

1. Személyre szabott szolgáltatások A mesterséges intelligencia segítségével szeretnénk a felhasználói élményt növelni, ezért személyre szabott ajánlatokkal, új funkciókkal bővítenénk, elemzés után építenénk be a projektbe.
2. Komplex utazási statisztikák A jövőben a

### Integrációs tervek

#### 1. Országos és nemzetközi rendszerintegráció

Országos közlekedési rendszerekkel való közvetlen kommunikáció

Multinacionális útvonaltervezési lehetőségek

Nemzetközi átszálló jegyek rendszere Egységes európai mobilitási platform

#### 2. Megosztott közlekedési eszközök integrációja

Kerékpáros és gyalogos útvonalak integrálása

Elektromos rollerek és megosztott járművek nyomon követése

Komplex mobilitási szolgáltatások összekötése

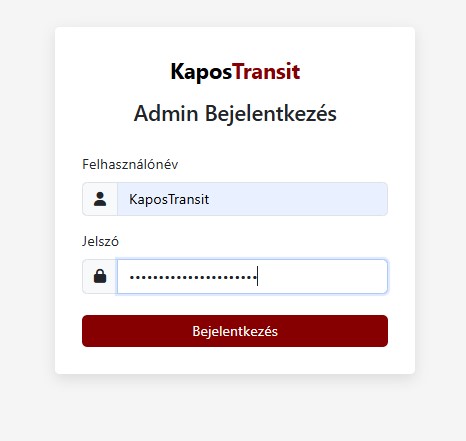
Multimodális közlekedési megoldások

## 12. Admin felület

A KaposTransit weboldalhoz készítettünk egy egyszerű admin felületet, ami segíti a késések és hírek kezelését. A rendszert PHP, MySQL és JavaScript technológiákkal valósítottuk meg. Ez az admin felület jelentősen megkönnyíti majd a jövőbeli admin felhasználók dolgát, akik a weboldal tartalmát fogják kezelni.

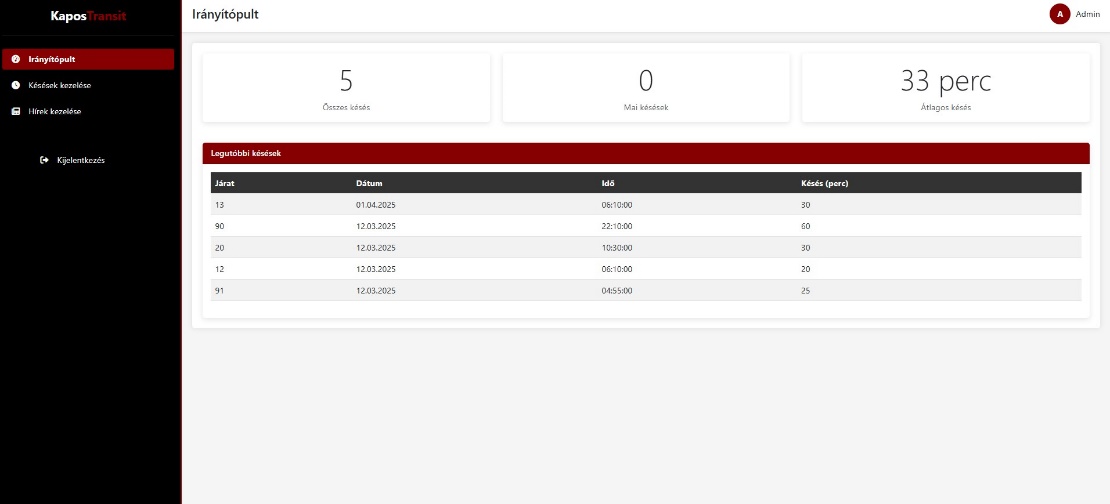
## 12.1 Bejelentkezési rendszer

Az admin felületre egy felhasználónévvel (KaposTransit) és jelszóval (KaposTransitAdmin997.@) lehet belépni. Sikeres bejelentkezés után láthatóvá válnak a kezelőfelületek, kijelentkezéskor pedig visszatérünk a bejelentkező oldalra.



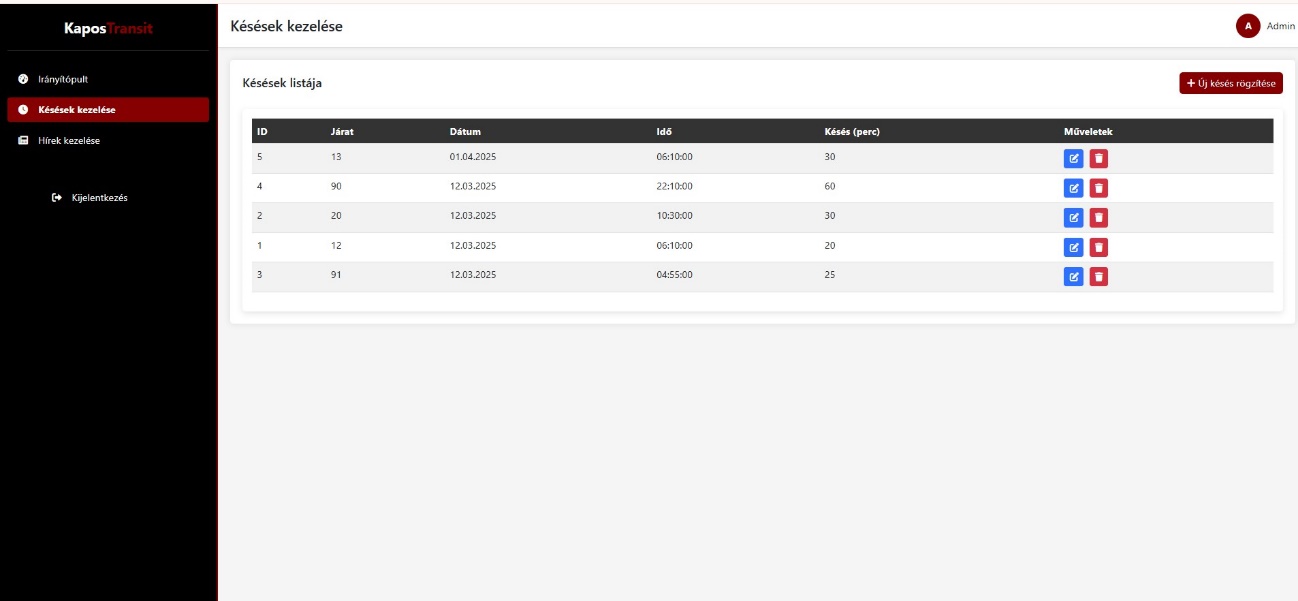
## 12.2 Irányítópult

Az irányítópulton láthatjuk a legfontosabb adatokat: összes késés száma, mai késések száma és átlagos késés időtartama. Egy táblázatban megjelennek a legutóbbi késések is. A felület Bootstrappel készült, így mobilon is jól használható.



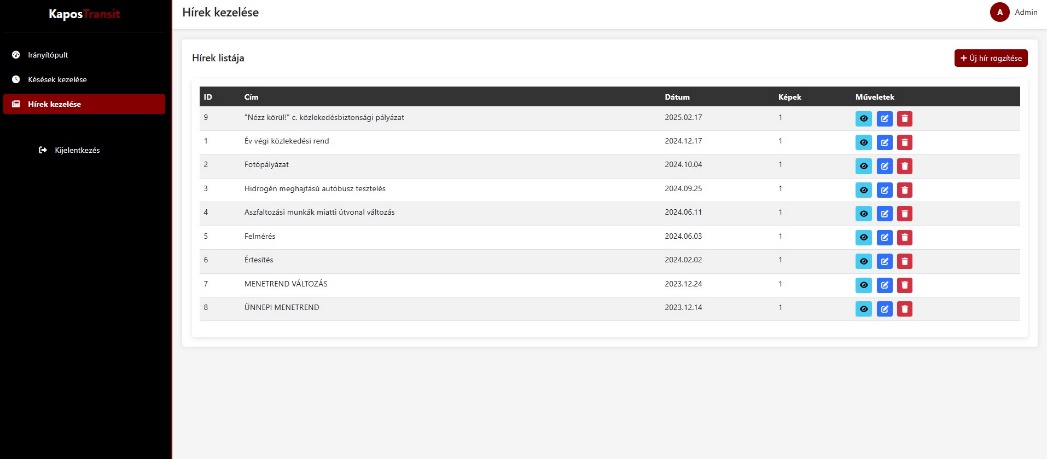
## 12.3 Késések kezelése

Ezen az oldalon kezelhetjük a járatkéséseket. Új késés rögzítéséhez meg kell adni a járat számát, dátumot, időpontot és a késés perceit. A rendszer csak akkor menti az adatokat, ha minden mező ki van töltve. A késéseket szerkeszthetjük és törölhetjük is, a műveletek a háttérben történnek az oldal újratöltése nélkül.



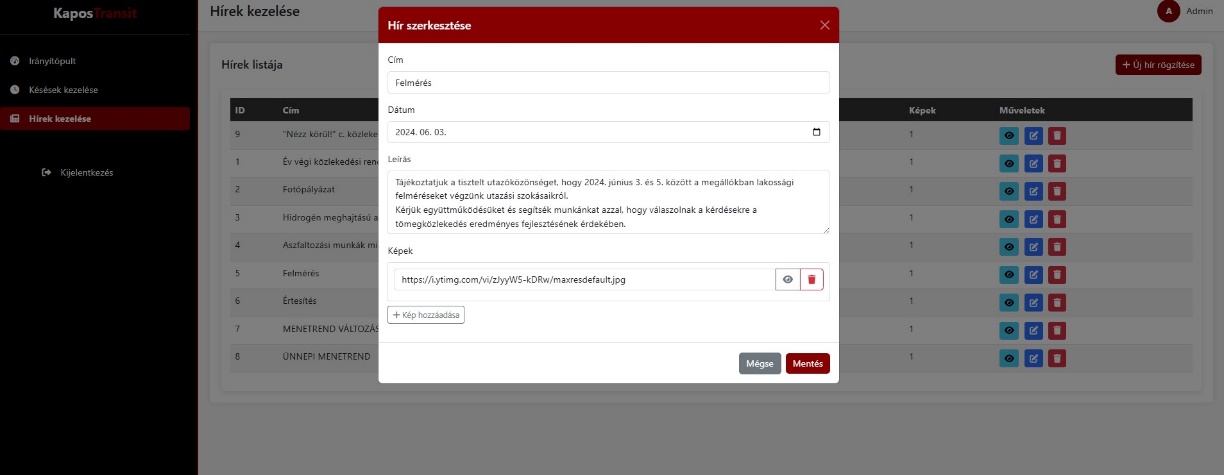
## 12.4 Hírek kezelése

Itt tudjuk kezelni a weboldal híreit és a hozzájuk tartozó képeket. Új hír létrehozásakor meg kell adni a címet, dátumot és a részletes leírást. Képeket is csatolhatunk a hírekhez URL-ek megadásával. A szerkesztés során módosíthatjuk az adatokat és a képeket, törlésnél pedig a kapcsolódó képek is törlődnek.



## 12.5 Az admin felület használata

A bal oldali menüben választhatunk a nézetek között. Új késés vagy hír létrehozásához a megfelelő oldalon a "+" gombra kell kattintani. A meglévő tartalmakat a sorok végén található gombok segítségével szerkeszthetjük vagy törölhetjük. A rendszer egyszerűen használható, különösebb informatikai tudás nélkül is.



## 12.6 Jövőbeli fejlesztések

A rendszert úgy alakítottuk ki, hogy a jövőben könnyen bővíthető legyen új funkciókkal. pl: megállók hozzáadása, törlése, módosításs jegyrendszer szabályozása, felhasználói adatok kezelése

## 13. Melléklet

### 13.1 Fogalomtár: Technikai és közlekedési kifejezések magyarázata

A KaposTransit használata során számos speciális technikai és közlekedési kifejezéssel találkozhat a felhasználó. Az átláthatóság és könnyebb értelmezhetőség érdekében összeállítottunk egy átfogó fogalomtárat.

### Technikai Kifejezések

1. GTFS (General Transit Feed Specification) A GTFS egy szabványosított formátum a tömegközlekedési menetrendek és kapcsolódó földrajzi adatok megosztására. Lehetővé teszi a különböző közlekedési rendszerek közötti adatcserét, nemzetközileg elfogadott szabvány.
2. API (Application Programming Interface) Szoftverek közötti kommunikációs felület, ami lehetővé teszi a különböző rendszerek adatcseréjét.
3. Reszponzív dizájn Olyan webes megjelenítési módszer, amely automatikusan igazodik a különböző eszközök képernyőméretéhez. Egységes felhasználói élményt biztosít minden eszközön.
4. Tokenizáció Bizalmas adatok biztonságos helyettesítése, személyes adatok védelme.
5. Valós idej**ű** adatfrissítés Folyamatos, azonnali adatszolgáltatás, naprakész információk biztosítása.

### Közlekedési kifejezések

Multimodális Közlekedés Több közlekedési módot magába foglaló utazási forma, különböző közlekedési eszközök kombinált használata, rugalmas és hatékony mobilitási megoldás.

13.2 Gyors hivatkozások: fontos elérhetőségek és linkek

### Hivatalos dokumentumok

A weboldal a Kaposbusz által létrehozott adatvédelmi szabályzatra, felhasználási feltételeire, általános szerződési feltételeire támaszkodik.

## **Záró gondolatok**

A KaposTransit több, mint egy egyszerű weboldal: olyan rendszer, ami segíti a kaposváriak mindennapi közlekedését, és folyamatosan fejlődik, hogy minél jobban megfeleljen az utazók igényeinek.