**PollákPizza**

A csapattagok: **Huszár Imre, Héja Gábor, Sztojka Milán László**

🛠️ Használt technológiák 🛠️

**Nyelvek: Vue.js, Node.js, CSS/SCSS**

✨ Design ✨

A Design tervezője**: Sztojka Milán**

Használt program: **Figma**

Tartalom

[Telepítési útmutató 2](#_Toc193783198)

[Frontend: 2](#_Toc193783199)

[Backend: 2](#_Toc193783200)

[Projekt struktúra 3](#_Toc193783201)

[Adatbázis 3](#_Toc193783202)

[Frontend 7](#_Toc193783203)

[Backend 13](#_Toc193783204)

[Tesztelés 21](#_Toc193783205)

[Hibakeresés 25](#_Toc193783206)

[Felhasználói dokumentáció 26](#_Toc193783207)

[Regisztráció 26](#_Toc193783208)

[Bejelentkezés 27](#_Toc193783209)

[Főoldal 27](#_Toc193783210)

[Admin Felület 28](#_Toc193783211)

[Menü oldal 29](#_Toc193783212)

[Rendelés felület 29](#_Toc193783213)

# Telepítési útmutató

Szükséges alapok:

A felhasználónak az alábbiakra van szüksége az projekt működéséhez:

* Egy működő számítógép amely windows 10-es operációs rendszert használ
* MySQL működését
* Node.js

## Frontend:

1. **Navigálj a mappába:**

* cd PollakPizza

1. **Telepítsd a szükséges csomagokat:**

* npm install

1. **Indítsd el a weboldalt:**

* npm run dev

## Backend:

1. **Navigálj a backend mappába:**

* cd PollakPizza/backend

1. **Telepítsd a szükséges csomagokat:**

* npm install

1. **Adatbázis feltöltése:**

* A MySQL-be hozzon létre egy adatbázist **pollakpizza** néven, majd importálja *docs*-ban található *PZdatabase.sql* fájlt

1. **Indítsd el a backendet:**

* node .

# Projekt struktúra

## Adatbázis

Az adatbázis felépítése:

1. pizzas tábla: A pizzák adatait tárolja.

|  |  |  |
| --- | --- | --- |
| Oszlop | Típus | Leírás |
| id | INT | Egyedi pizza azonosító |
| name | VARCHAR(255) | A pizza neve (pl. Margherita) |
| price | INT | A pizza alapára. |
| image | VARCHAR(255) | URL útvonal a pizza képéhez. |

2. size tábla: A pizzaméreteket és az árnövekményeket tartalmazza.

|  |  |  |
| --- | --- | --- |
| Oszlop | Típus | Leírás |
| id | INT | Egyedi méret azonosító |
| size | INT | A pizza mérete (pl. 32). |
| multiPrice | Decimal(10,0) | Méret alapján árnövekmény. |

3. toppings tábla: Az elérhető feltéteket és azok többletköltségeit tárolja.

|  |  |  |
| --- | --- | --- |
| Oszlop | Típus | Leírás |
| id | INT | Egyedi feltét azonosító |
| name | VARCHAR(255) | A feltét neve (pl. Sajt). |
| bonusPrice | INT | A feltétért járó többletköltség |

4. users tábla: A felhasználók adatait tárolja.

|  |  |  |
| --- | --- | --- |
| Oszlop | Típus | Leírás |
| id | INT | Egyedi felhasználói azonosító |
| name | VARCHAR(255) | A felhasználó teljes neve |
| email | VARCHAR(255) | A felhasználó email címe |
| password | VARCHAR(255) | A felhasználó jelszava (titkosított) |
| birthdate | DATE | A felhasználó születési dátum |
| address | VARCHAR(255) | Kiszállítási cím. |
| admin | TINYINT | Admin státuszt jelöl (0/1). |
| phonenumber | VARCHAR(50) | Telefonszám |
| resetPasswordToken | VARCHAR(255) | Jelszó visszaállítási token |
| resetPasswordExpires | BIGINT | A visszaállító token lejárati ideje |

5. orders tábla A leadott pizzarendeléseket kezeli.

|  |  |  |
| --- | --- | --- |
| Oszlop | Típus | Leírás |
| id | INT | Egyedi rendelési azonosító |
| userId | INT | Idegen kulcs a users táblára |
| pizzaId | INT | Idegen kulcs a pizzas táblára |
| pizzaNum | INT(11) | A rendelési pizza darabszáma |
| sizeId | INT | Idegen kulcs a size táblára. |
| address | VARCHAR(255) | A rendelés kiszállítási címe |
| userPhone | VARCHAR(50) | A rendelő telefonszáma |
| finalPrice | INT | A végleges ár |
| status | VARCHAR(50) | A rendelés státusza (pl. Kész). |

6. orderTops tábla: A rendeléshez tartozó feltéteket tárolja. Több a Többhöz kapcsolat az orders és a toppings táblák között.

|  |  |  |
| --- | --- | --- |
| Oszlop | Típus | Leírás |
| orderId | INT | Idegen kulcs az orders táblára |
| toppingId | INT | Idegen kulcs a toppings táblára. |

7. pizzaToppings tábla A pizzákhoz tartozó feltéteket tartalmazza. Több a Többhöz kapcsolat a pizzas és a toppings táblák között.

|  |  |  |
| --- | --- | --- |
| Oszlop | Típus | Leírás |
| pizzaId | INT | Idegen kulcs a pizzas táblára |
| toppingId | INT | Idegen kulcs a toppings táblára. |

Kapcsolatok Egy-Többhöz kapcsolatok:

orders → users

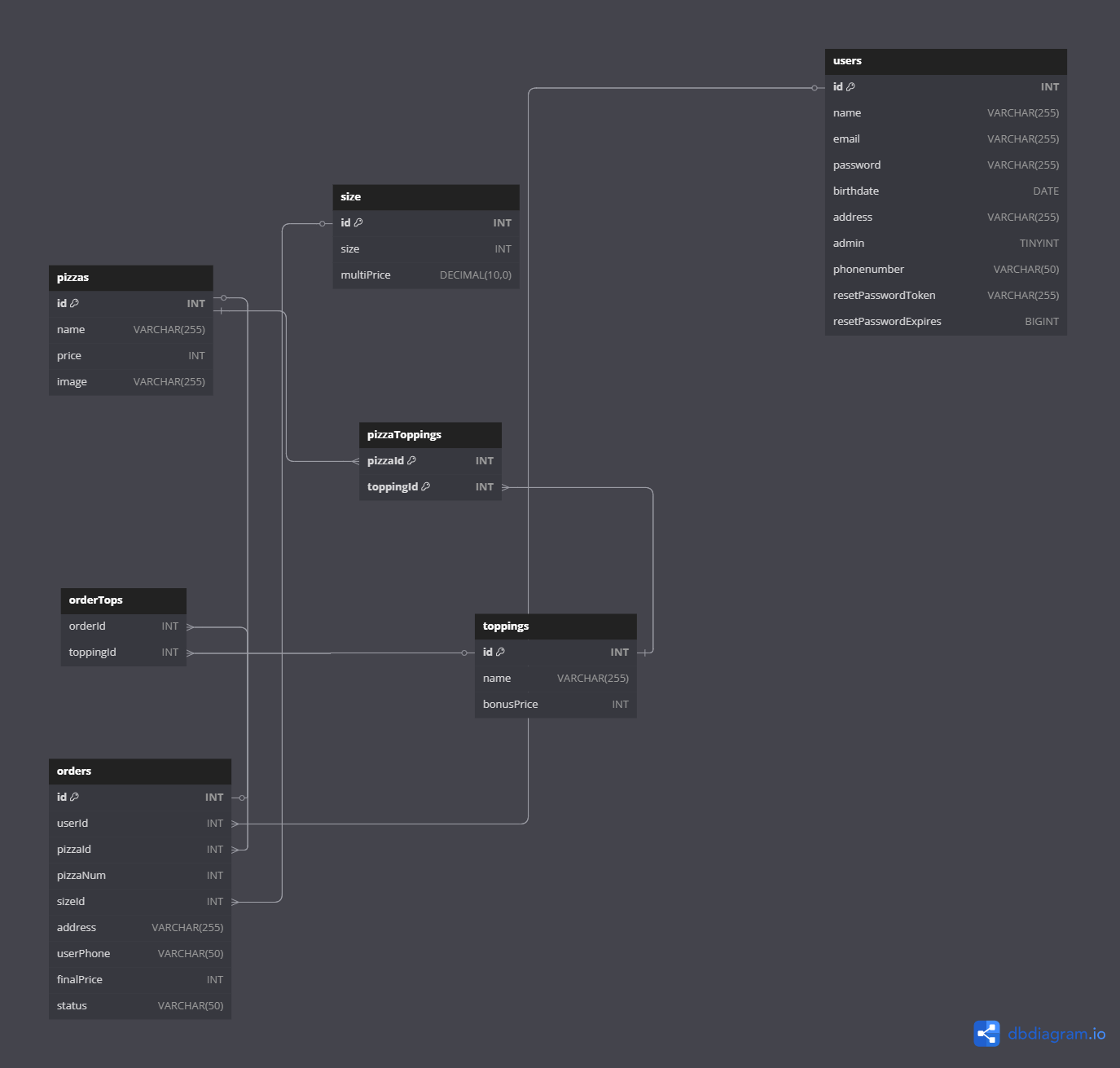
orders → size

orders → pizzas

Több-Többhöz kapcsolatok:

pizzas ↔ toppings a pizzaToppings táblán keresztül.

orders ↔ toppings az orderTops táblán keresztül.



Alapértelmezett Adatok Előre definiált pizzák:

|  |  |  |
| --- | --- | --- |
| Név | Ár(FT) | Kép URL |
| Margherita | 1000 | src/assets/image/margareta.jpg |
| Pepperoni | 1200 | src/assets/image/pepi.jpg |
| Hawaii | 1300 | src/assets/image/hawaii.jpg |

Előre definiált feltétek:

|  |  |
| --- | --- |
| Név | Ár(FT) |
| Sajt | 200 |
| Pepperoni | 300 |
| Ananász | 250 |
| Sonka | 300 |

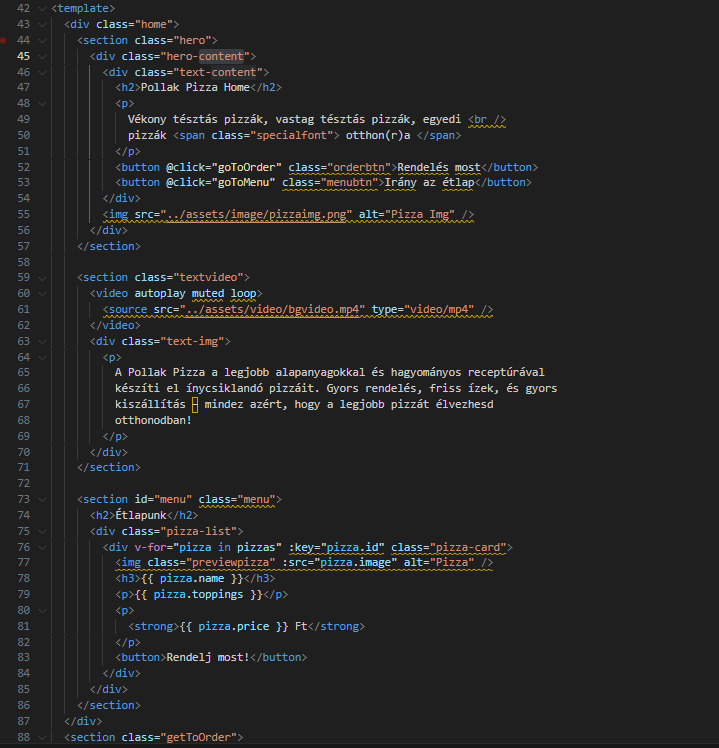
Pizza-Feltét Kapcsolatok:

|  |  |
| --- | --- |
| Név | Ár(FT) |
| Margherita | Sajt |
| Pepperoni | Sajt, Pepperoni |
| Hawaii | Sajt, Ananász, Sonka. |

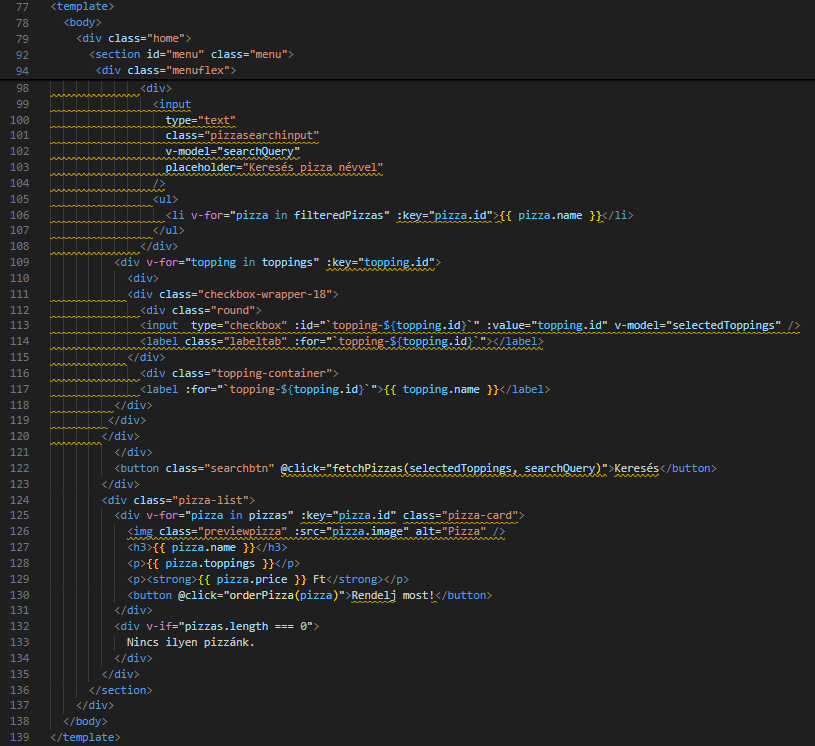
## Frontend

A frontend Vue.js alapú, és a következő főbb komponensekből áll(A képekben nincs benne mindig a teljes kód):

**HomePage.vue:** A főoldal, ahol a felhasználók megtekinthetik az étlapot és rendelhetnek pizzát.

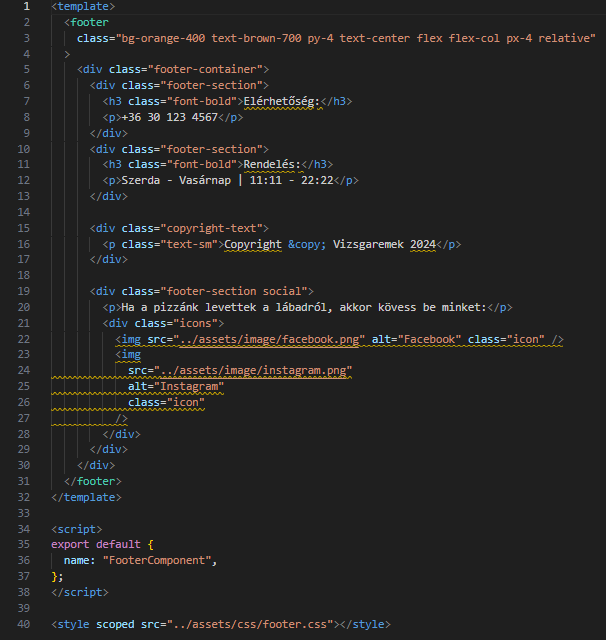


**MenuPage.vue:** Az étlap oldal, ahol a felhasználók böngészhetik a pizzák listáját és rendelhetnek.



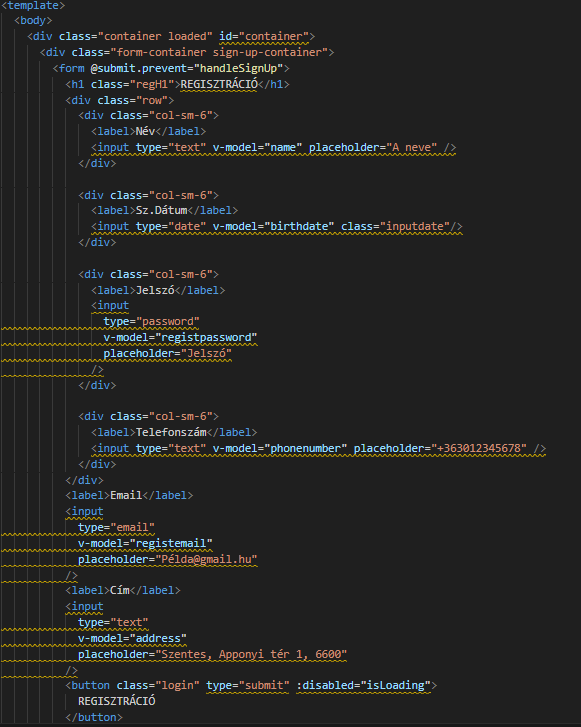
**AdminPage.vue:** Az Admin felület, itt lehet majd a pizzákat és a felhasználókat kezelni.

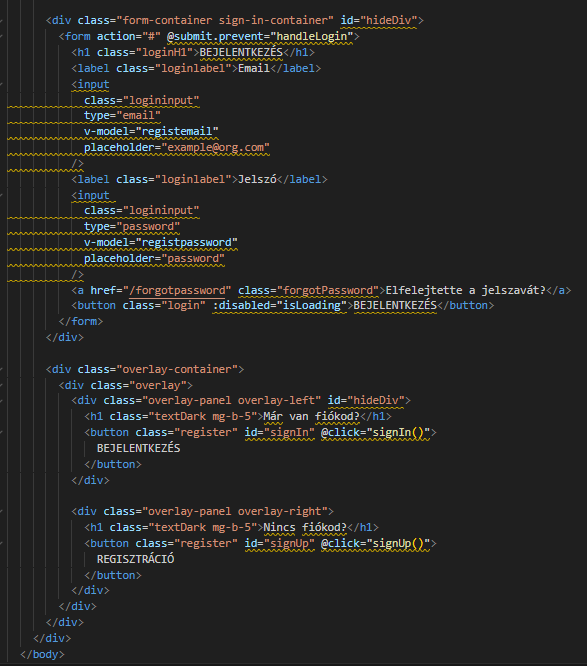
**Footer.vue:** Az oldalak által használt footer.



**ForgotPasswordPage.vue:** Az oldal melyen képes megadni a regisztrált email címét és küldünk önnek a jelszó helyreállító linket

**LoginPage.vue:** A első oldal, amit lát a felhasználó amikor az oldalra lép, itt képes a felhasználó egyszerre belépni, és regisztrálni is.





**Navbar.vue:** A felső navigációs sáv, használatával képes az oldalaink közt a felhasználó mozogni.

**OrderPage.vue:** Az oldal melyen képes a felhasználó rendelést leadni a weboldalon.

**OrdersPage.vue:** Ezen az oldalon képesek lesznek az adminok és dolgozók megfigyelni a bejövő rendeléseket és azoknak státuszát változtatni

**ProfilePage.vue:** Ezen az oldalon a felhasználó, ha szeretné változtathat meg számos adatot melyet mi tárolunk a bejelentkezés, és kiszállítás gyorsaságának érdekében.



**ResetPasswordPage.vue:** Ezt az oldalt a felhasználó abban az esetben tekinti meg, ha valamilyen módon elfelejtette jelszavát, az emailben megkapott linkre kattintva itt meg tudja változtatni.

## Backend

A backend Node.js és Express alapú, és a következő főbb részekből áll(A képekben nincs benne mindig a teljes kód):

**app.js:** Az Express alkalmazás inicializálása és a fő útvonalak beállítása.



**routes:** Az alkalmazás különböző útvonalai.

*addPizzas.js:* Pizza hozzáadására, és feltétek hozzáadására szolgál

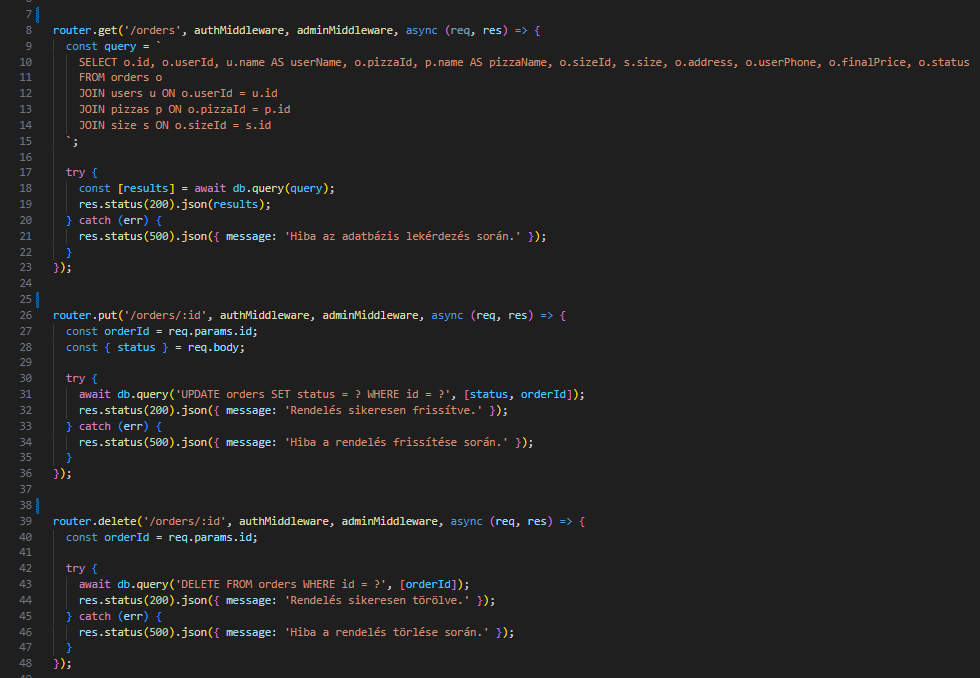


*admin.js:* A felhasználók, és a rendeléseik lekérdezését kezeli.

*allpizzas.js:* Lekérdez hat véletlenszerű pizzát

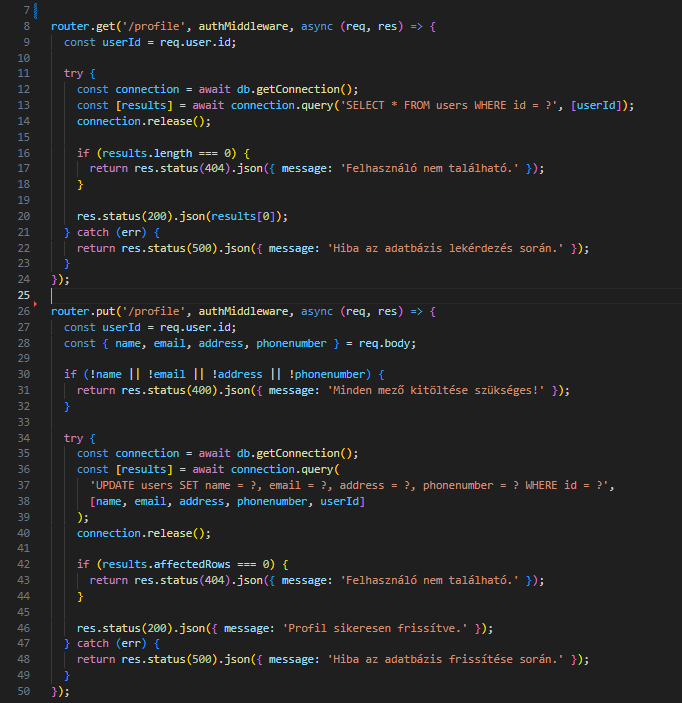
*allsizes.js:* Lekérdezi az összes méretet

*orders.js:* A rendeléseket kezeli, lekérdez, frissít, kész rendelést töröl, feladni rendelést.



*auth.js*: Bejelentkezési és regisztrációs útvonalak.

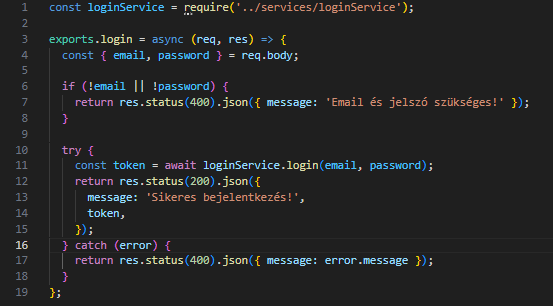
*profile.js*: Felhasználói profil útvonalak.



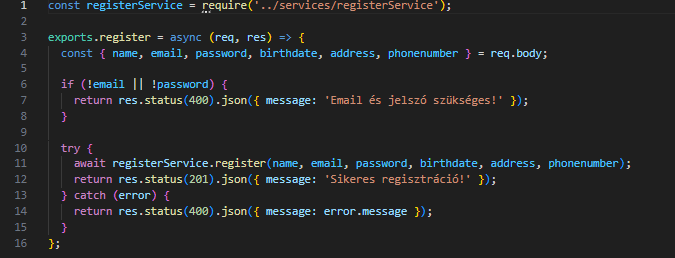
*pizzas.js*: Pizzák listájának lekérdezése.

**controllers**: Az útvonalakhoz tartozó vezérlők.

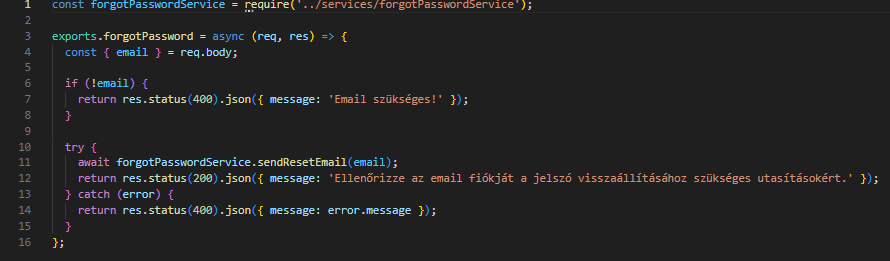
*loginController.js:* Bejelentkezési logika.



*registerController***.***js***:** Regisztrációs logika.



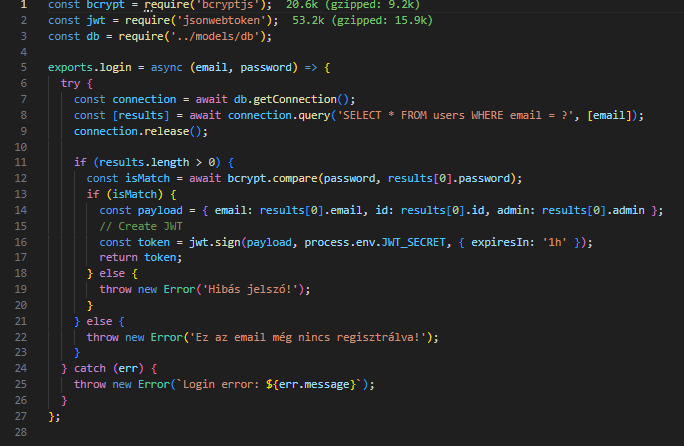
*forgotPasswordController.js:* Kezeli az emailt.



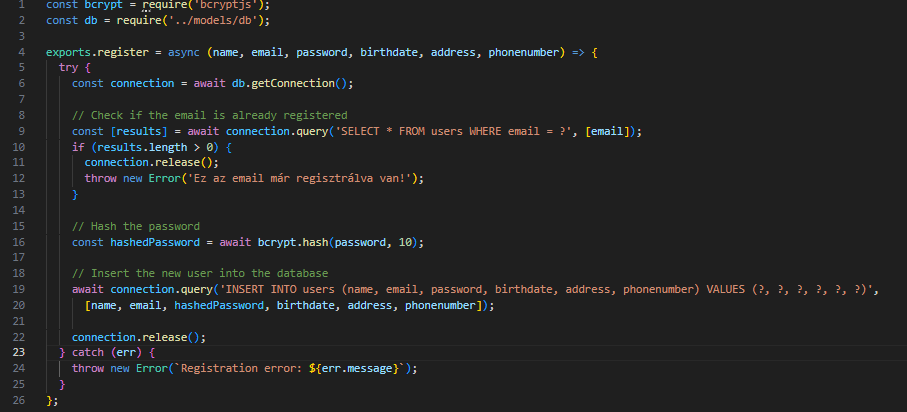
*resetPasswordController.js:* Kezeli, hogy a jelszót újra lehessen írni.

**services:** Az üzleti logika és adatbázis műveletek.

*loginService.js***:** Bejelentkezési szolgáltatás**.**



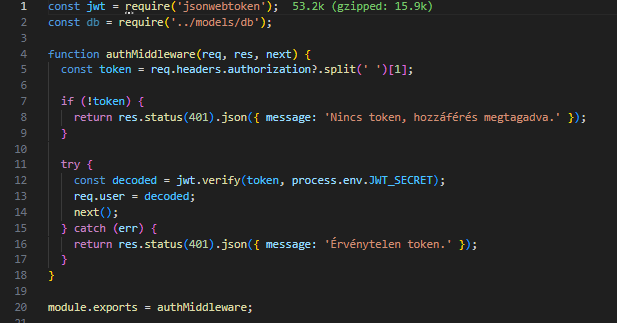
*registerService.js:* Regisztrációs szolgáltatás.



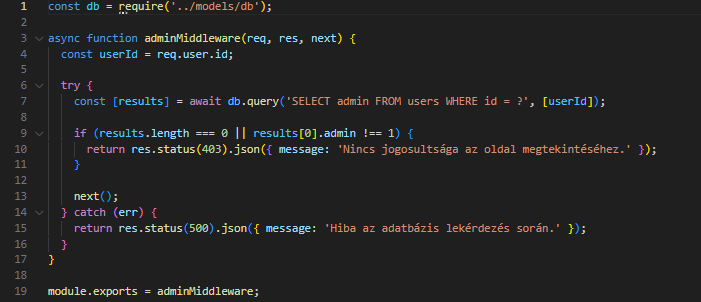
*forgotPasswordService.js:* Itt nézi meg hogy az email helyes-e, majd elküldi az emailt, ha helyes az email.

**middleware:** Köztes rétegek, például hitelesítés.

*auth.js:* JWT token hitelesítés.



*admin.js:* Megfigyeli, hogy a felhasználó admin-e vagy nem



**models**: Az adatbázis modellek.

*db.js***:** MySQL adatbázis kapcsolat beállítása.



## Tesztelés

A projekt teszteléséhez a Jest és supertest könyvtárakat használjuk.

Tesztfájlok

A tesztfájlok a tests mappában találhatók:

**authRoutes.test.js:** Tesztek a bejelentkezési és regisztrációs útvonalakhoz.

**profileRoutes.test.js:** Tesztek a felhasználói profil útvonalakhoz.

**pizzaRoutes.test.js:** Tesztek a pizzák listájának lekérdezéséhez.

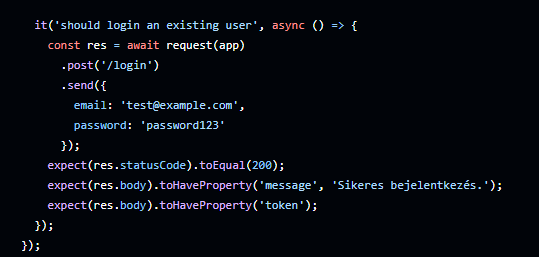
Tesztek futtatása

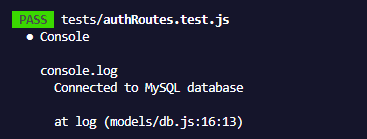
A tesztek futtatásához használd a következő parancsot a terminálban: **npm test**

Példák

Bejelentkezés

A bejelentkezési útvonal a /login végponton érhető el. A kérésnek tartalmaznia kell az email címet és a jelszót.

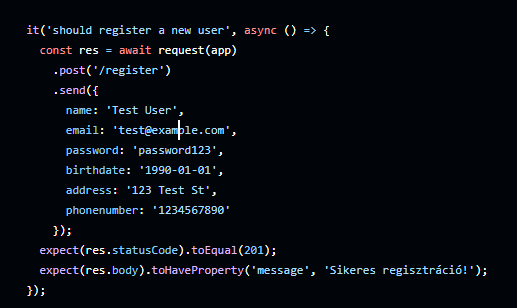
Példa kérés: 

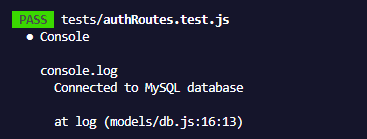


Regisztráció

A regisztrációs útvonal a /register végponton érhető el. A kérésnek tartalmaznia kell a felhasználó nevét, email címét, jelszavát, születési dátumát, címét és telefonszámát.

Példa kérés:

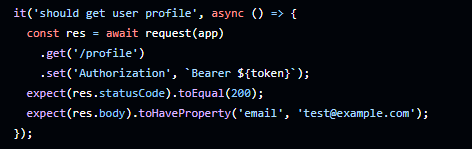


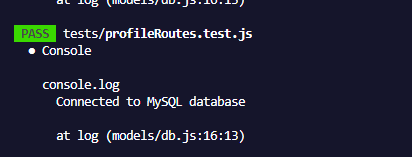


Profil lekérés

A felhasználói profil lekérdezése a /profile végponton érhető el.

Példa kérés:

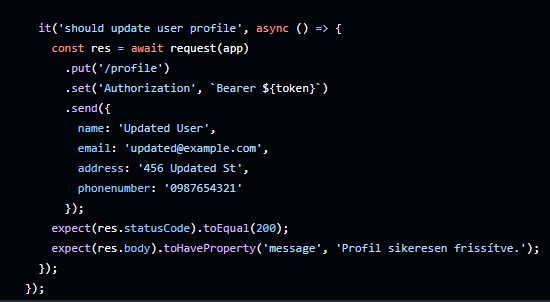


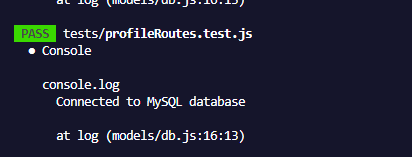


Profil frissítése

A felhasználói profil frissítése a /profile végponton érhető el. A kérésnek tartalmaznia kell egy érvényes JWT tokent az Authorization fejlécben, valamint a frissítendő adatokat.

Példa kérés:

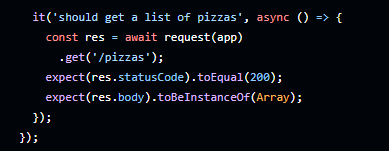


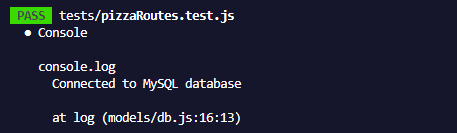


Pizzák lekérése

A pizzák listájának lekérdezése a /pizzas végponton érhető el.

Példa kérés:





## Hibakeresés

Ha problémák merülnek fel a tesztek futtatása során, ellenőrizd a következőket:

Győződj meg róla, hogy az **adatbázis elérhető** és **megfelelően van beállítva**.

Ellenőrizd, hogy a **szükséges csomagok telepítve** vannak.

Győződj meg róla, hogy a tesztadatok **megfelelően vannak beállítva** és tiszták a tesztek futtatása előtt.

# Felhasználói dokumentáció

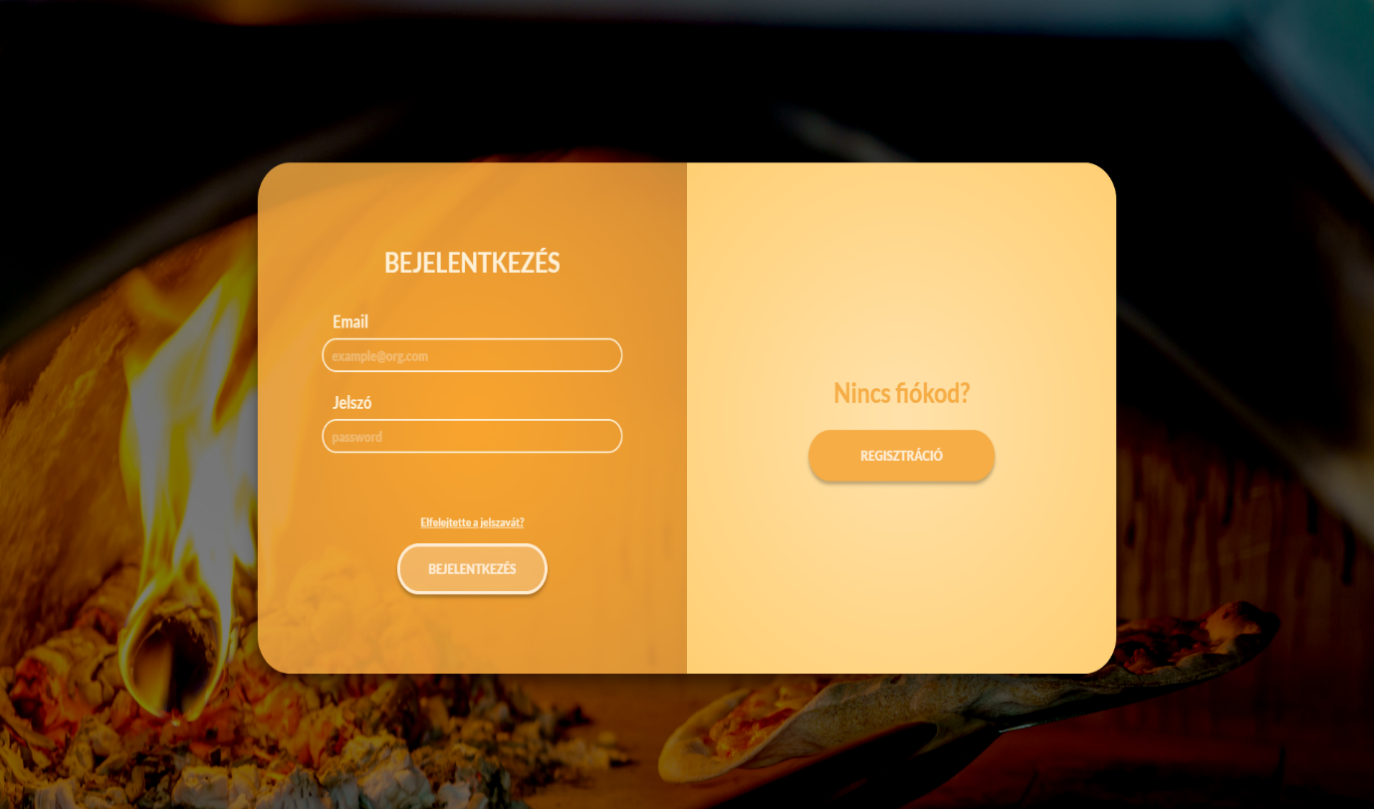
Ez a dokumentum bemutatja hogyan képes használni az oldalunkat

## Regisztráció

Kezdésként a bejelentkezés oldalt fogja látni, ahhoz hogy regisztráljon rá kell nyomnia a "REGISZTRÁCIÓ" gombra, ez majd átirányítja önt. Miután regisztrációt csinálja a "Név" mezőhöz írja be a nevét, a születési dátumnál a naptár iconra kattintva kiválaszthatja a dátumot vagy be is írhatja. A "Jelszó"-nál adjon meg egy jelszavat amelyet mindig használni fog a bejelentkezésnél. "Telefonszám"-nál adja meg a telefonszámát. Az "Email"-hez adja meg az emailét, és a végén a "Cím"-hez pedig a címét véglegesen. Miután végzett adatainak megadásával kattintson a "REGISZTRÁCIÓ" gombra. Egy kis visszajelzéssel mutatjuk önnek ha sikeres a regisztráció.



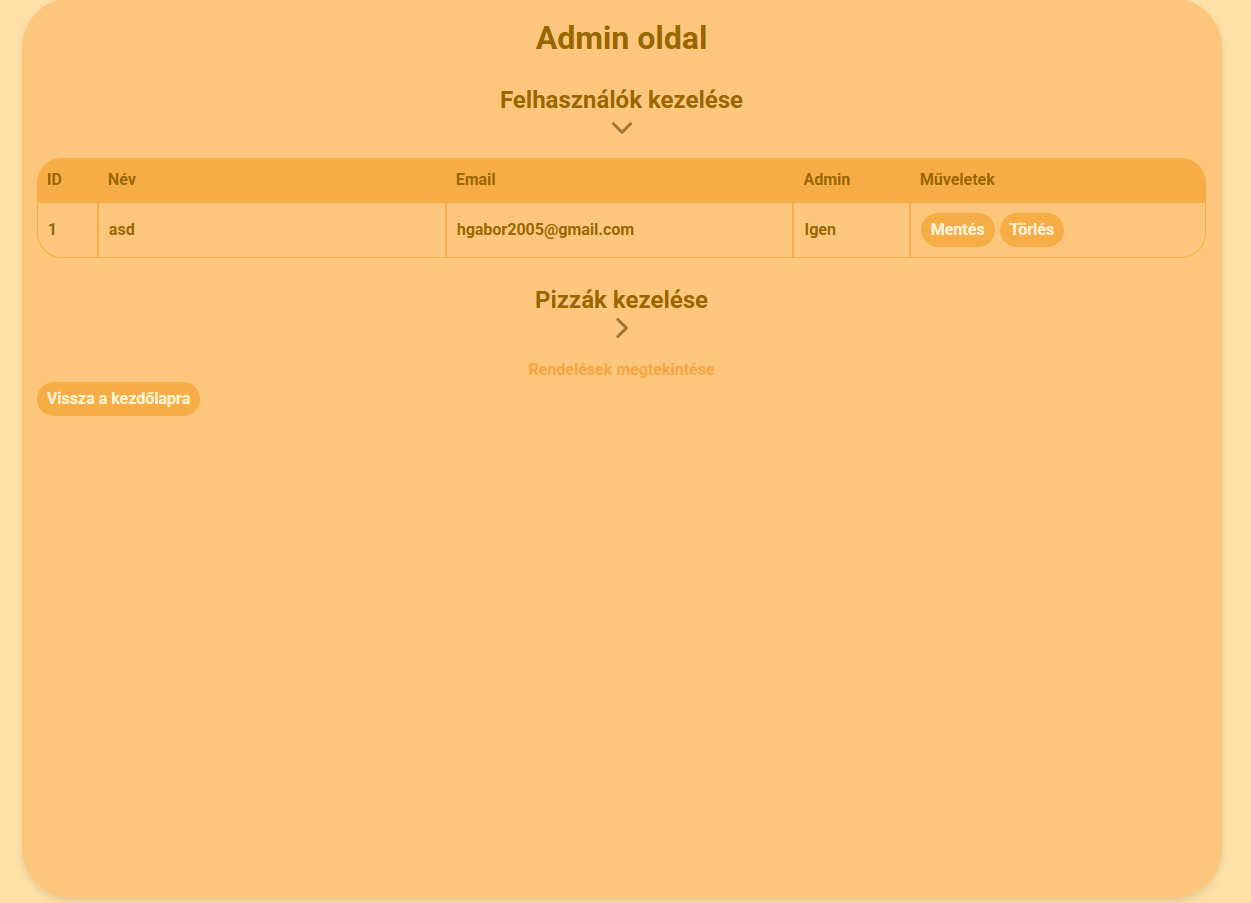
## Bejelentkezés

Miután a felhasználó sikeresen regisztrált (vagy már rendelkezett egy felhasználóval), képes lesz bejelentkezni. A bejelentkezéshez szükség van az email címre amivel regisztrált, és a jelszavára. Abban az esetben ha elfelejtette a jelszavát rád tud menni a "Elfelejtette a jelszavát" ahol be írja a email címet amivel regisztrált, ezután ha helyesen írta be akkor a email címére küldünk egy emailt amiben van egy link amelyre rákattintva átvezet egy oldalra ahol a jelszavát meg tudja változtatni, miután megváltoztatta abban az esetben újra be tud lépni.

## Főoldal

Ez az oldal amilyet elsőre lesz képes megtekinteni miután belépett, innen képes elérni a szolgáltatásaink nagy részét, ha lejjebb görget képes megtekinteni pizza ajánlatokat, illetve el tud kezdeni rendelni vagy az oldal alján található gomb lenyomásával vagy a navigációs sávon rákattint a "Rendelés" menüpontra, vagy képes elérni a pizzánk listáját ahol képes megtekinteni az összes pizzát amivel rendelkezünk, innen képes lesz azt a pizzát rendelni azonnal vagy kosárba helyezni hogy tovább folytassa a vásárlását.

## Admin Felület

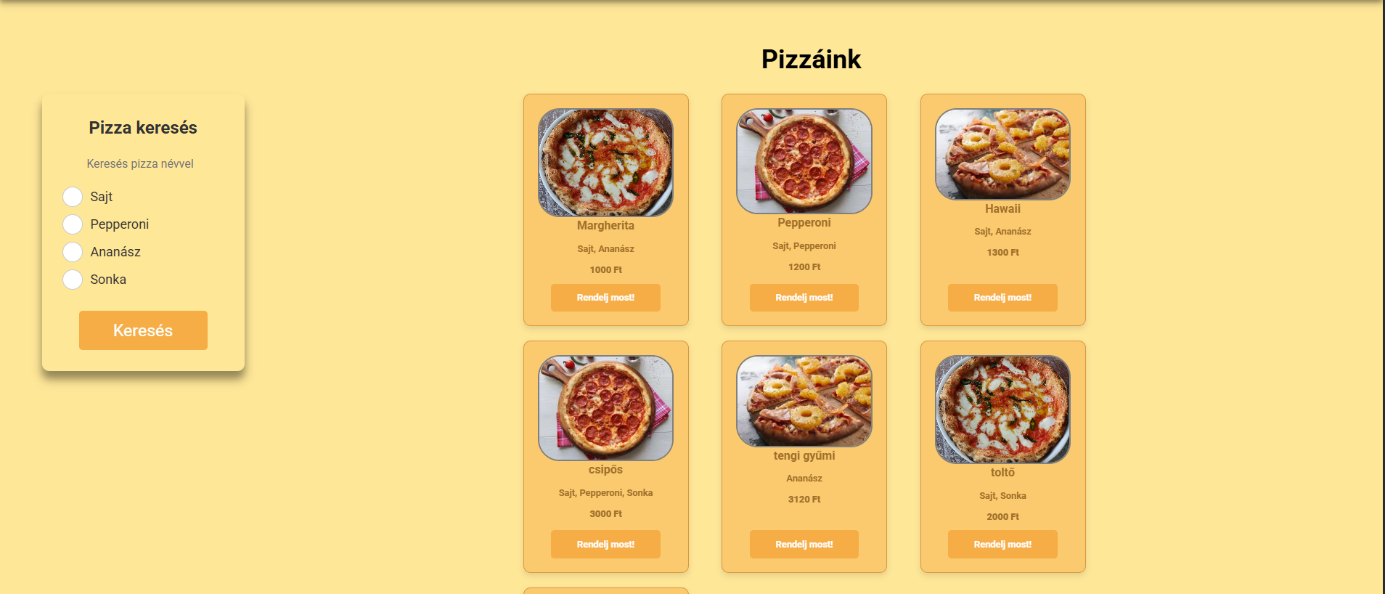
****Az admin felület az amelyben képes a felhasználókat kezelni és a menüben elérhető pizzákat. Az oldal két részre van osztva, a felhasználó kezelés és a pizza kezelés. A felhasználó kezelésnél képes a admin törölni, illetve változtatásokat végrehajtani a felhasználó adatain. A másik pont a pizzák kezelése, itt lehet megtekinteni a jelenlegi pizzákat, azoknak árát, feltéteket amik vannak az adott pizzán, és képét annak a pizzának. Az alatt van egy felület amelyet egy gomb segítségével lehet megjeleníteni. A gombra nyomva megjelennek a beviteli mezők amivel meg tudja az új pizza nevét, alapárát, képét, és feltéteket rajta.



## Rendelés kezelés oldal

****Ez az oldal szolgál arra hogy megtekintsük a jelenlegi rendeléseinket, ezeket az információkat csak az alkalmazottak és adminok érik el. Meg tudnak majd jelölni rendelést úgy mint ”Kész”, ha megjelöli átlép egy másik részbe a ”Kész rendelésekbe”, innen ha már nem kívánjuk látni akkor képesek vagyunk törölni a rendelést.

## Menü oldal

A menü oldal arra szolgál hogy a felhasználó megtekintse a rendelkezésre álló pizzáinkat, ennek az oldalnak a segítségével akár meg is kereshet olyan pizzát amilyet preferál hogy a rendelésnél tudja a felhasználó milyen pizzát szeretne rendelni. Pizzát lehet keresni név és feltét alapján is, természetesen ha a neves keresést választja akkor megkeresi az összes pizzát ami hasonló névvel rendelkezik mint a keresésben megadott név, illetve feltét alapján is amivel meg tudja adni milyen feltétek vannak az adott pizzák amiket ön keres. 

## Rendelés felület

A rendelési felület szolgál arra hogy a felhasználó képes legyen rendelést létrehozni. Ezen az oldalon lesz képes a kosárba be tenni pizzákat a felhasználó és itt tudja változtatni a pizzák méretét. A kiválasztott pizzát a kosárba tesszük ahol tudja növelni és csökkenteni hogy mennyit kíván az adott pizzából rendelni, esetleg más méretben akarja azt a pizzát vagy más pizzát akar azt is meg tudja oldani a felhasználó. Képes a felhasználó saját ízléséhez csinálni egyedi pizzát is, így biztosítva ha nem talál a felhasználó olyan pizzát amit szeretne a menüről rendelni akkor is tudjuk megfelelő pizzát létrehozni.

A rendelés felület amit a felhasználó lát belépésnél

A rendelés miután a felhasználó választott pizzát



A gombra kattintva megjelenő fizető felület

Egyedi pizza rendelő felület