



## FoodHub recept weboldal

A Foodhub egy recept weboldal, ahol böngészéssel és kereséssel megtalálható számos ételrecept. Felhasználók regisztrálhatnak, bejelentkezhetnek, kedvenc receptjeiket elmenthetik és megoszthatják másokkal. Emellett lehetőségük is van saját receptek feltöltésére is.

Budai Bence  
Tüdős Levente

## Tartalomjegyzék

Bevezetés: .....	3
Felépítés / Funkciók: .....	3
Követelményanalízis: .....	4
Tervezés: .....	5
Fejlesztés és implementáció: .....	5
Projektmenedzsment:.....	6
Záró megjegyzések és köszönetnyilvánítás: .....	7
A projekt munka telepítése: .....	8
Szükséges alkalmazások:.....	8
Projektmunka telepítése:.....	8
Az adatbázis felépítése: .....	8
A weboldal megnyitása:.....	8
Weblapok: .....	9
Főoldal:.....	9
Bejelentkezés: .....	9
Regisztráció: .....	10
Recept kategória: .....	11
Profil:.....	11
Recept feltöltés: .....	12
Végpontok:.....	15
Felhasználó:.....	15
/userRecipes/:id .....	15
/user/:id .....	15
/regUser .....	15
/updateUser .....	16
/login .....	16
Recept: .....	17
/recipe/:id .....	17
/uploadRecipe .....	17
/getComment/:id .....	18
/uploadComment.....	18
/getFavorites/:id .....	19
/addFavorite.....	19
/removeFavorite .....	20
/ingredients.....	20
/upload.....	21
/notVerifiedRecipes .....	21

/verifyRecipe .....	22
/deleteRecipe .....	22
/giveAdminRole .....	23
/deleteAdminRole .....	23
/getAdmins .....	24
/uploadIngredient .....	24
/updateIngredient .....	25
/removeIngredient .....	25

# FoodHub recept weboldal

## Bevezetés:

A FoodHub weboldalunkat úgy terveztük meg hogy minden korosztály tudja könnyedén használni és minden étvágy gerjesztőbb legyen a dizájn.

A weboldal nincsen a végleges fázisban mivel sok ötletünk lett volna még hogy hogyan tudnánk még érdekesebbé, korszerűbbé vagy még kiemelkedőbe tenni, mint a többi recept weboldalt az interneten. Persze mind a ketten a csapat társammal törekedtünk arra, hogy a legjobb tudásunkat felhasználva el tudjuk készíteni ezt a weboldalt. A jövőben szeretnénk tovább fejleszteni és akár még profitálni is belőle.

## Felépítés / Funkciók:

A weboldal felépítésénél sok mindenre törekedtünk. Számunkra a legfontosabb az átláthatóság volt mivel tudjuk hogyha egy weboldal nincsen rendesen felépítve akkor az nem fogja felkelteni senkinek a figyelmét.

A legfőbb funkciók, amiket létrehoztunk a weblapon:

- Receptgyűjtemény böngészése:  
A felhasználók szabadon böngészhetnek a receptgyűjteményben, kategóriák, ételtípusok vagy diétás preferenciák szerint szűrve.
- Keresés:  
A felhasználók könnyen megtalálhatják az általuk keresett recepteket a keresőmező segítségével, kulcsszavak vagy recept neve alapján.
- Receptek megtekintése:  
A weboldalon minden recept részletes leírással, hozzávalók listájával, elkészítési lépéseivel és esetleges képekkel jelenik meg.
- Receptek értékelése és vélemények megosztása:  
A felhasználók értékelhetik és megjegyzéseikkel megoszthatják tapasztalataikat a kipróbált receptekről.
- Felhasználói profilok:  
Regisztrált felhasználók létrehozhatnak profilokat, ahol menthetik kedvenc receptjeiket, és követhetik az általuk kedvelt konyhaművészeket vagy barátokat.
- Kérdések és válaszok:  
Felhasználók kérdezhetnek és válaszolhatnak egymásnak a receptekkel vagy a főzéssel kapcsolatos kérdéseikre

Célunk ezekkel a funkciókkal, hogy a felhasználók tudjanak hatékonyan keresni vagy kommentálni a receptekhez így segítve egymást a konyhában.

## Követelményanalízis:

- **Életszerű, valódi problémára nyújt megoldást?**
  - Szerintünk ezt a weboldalt lehet a mindennapokban használni mivel sokan szeretnek főzni és nem mindenki szereti a régi recept könyveket bújni, hogy el tudjon készíteni egy finom ételt.
- **Adattárolási és -kezelési funkciókat is megvalósít. Az adatbázis legalább 5 táblát és triggeret, tárolt eljárásokat függvényeket tartalmaz.**
  - Ezt szintén teljesítettük hiszen a receptek tárolására MySQL adatbázist használtunk
- **RESTful architektúrának megfelelő szerver és kliens oldali komponenseket egyaránt tartalmaz.**
  - A kliens oldal tökéletesen tud kommunikálni a szerver oldallal hiszen enélkül nem tudna működni a weboldal.
- **A kliens oldali komponens vagy komponensek egyaránt alkalmasak asztali és mobil eszközökön történő használatra.**
  - A weboldalunk teljesen reszponzív, így a hétköznapi eszközökön mindenki tudja használni, akár asztali számítógépen, laptopon, tableten vagy telefonon keresné fel weboldalunkat.
- **A forráskódnak a tiszta kód elveinek megfelelően kell készülnie.**
  - A tisztakód elveit betartva kódoltunk minden fájlt, hiszen a kódjain egyszerűek, de még is tömörek, minden egyes részt kommentekkel ellátva lehet megtalálni hogy még az is értse aki csak beleszeretne nézni hogy még is mi ez, a funkciók önmagukban is működnek de némelyikek egymásra támaszkodnak hogy végre tudják hajtani a nekik szánt feladatokat, a függvények értelmesen céljukra utaló nevekkkel vannak ellátva, a tesztelhetőség kifejezetten könnyen elvégezhető hiszen mi megírtunk pár tesztet előre hogy ezzel is segítsük az áttekinthetőséget.
- **A szoftver célját, komponenseinek technikai leírását, működésének műszaki feltételeit és használatának rövid bemutatását tartalmazó dokumentáció is része a csomagnak.**
  - Ez a dokumentum pontosan erre szolgál.
- **Biztonsági technológiákat alkalmaz az adatbázis RESTapi-k és az adminisztrációs felület stb., használata során.**
  - A biztonsághoz nálunk a jelszók hash-elése tesz ennek eleget, mivel egy olyan módszert használunk, amit nem lehet feltörni mivel nem lehet visszaalakítani a már átalakított jelszavakat.

## Tervezés:

- A FoodHub weboldal tervezése során részletesen kidolgoztuk a felhasználói felületet és a weboldal architektúráját. Ez magában foglalja az adatbázis-séma tervezését, az adatmodellt és az alkalmazás rétegzettségét.
- A tervezés során figyelembe vettük a felhasználói élményt és az intuitív navigációt, hogy a weboldal használata könnyű legyen mindenki számára.
- Emellett gondoskodunk a weboldal megjelenéséről és általános esztétikájáról, hogy vonzó és felhasználóbarát legyen.
- A tervezési folyamat fontos része volt a felhasználói visszajelzések és tesztek integrálása annak érdekében, hogy a végeredmény minőségi és hatékony legyen.
- A projekt megfelelő architektúrájának kiválasztása az egyik legfontosabb lépés a tervezés során. Az architektúra olyan alapvető keretet biztosít, amelyben a kód strukturált és könnyen karbantartható lesz.
- Az általános kódolási stílus és konvenciók betartása segített abban, hogy a kód egységes legyen és könnyen olvasható más fejlesztők számára is. Ez magában foglalhatja az elnevezési konvenciókat, az elrendezési szabályokat és más formai irányelveket, mint például a folyamatos kommentelést, jegyzetelést.

## Fejlesztés és implementáció:

- A FoodHub weboldal fejlesztése és implementációja során különböző technológiai eszközöket és platformokat használunk fel.
- Használt alkalmazásaink:
  - **Visual Studio Code (VS Code):** Ez egy nagyon népszerű és kiterjeszthető kódszerkesztő, amely támogatja a különböző programozási nyelveket és integrációkat, mint például a JavaScript, HTML, CSS, valamint különféle kiegészítőket és fejlesztői eszközöket kínál.
  - **XAMPP:** Ez egy nyílt forráskódú szerver alkalmazás, amely lehetővé teszi a PHP, MySQL, Apache és más szoftverek telepítését és futtatását. Különösen hasznos volt a weboldal fejlesztésében és tesztelésében a lokális környezetben.
  - **Node.js:** Ez egy JavaScript futtatókörnyezet, amely lehetővé teszi a JavaScript kód futtatását a szerveroldalon. Node.js segítségével lehetőség van aszinkron programozásra, moduláris fejlesztésre és a különféle csomagok (package) használatára, amelyek széleskörű funkcionalitást biztosítottak.
  - **HTML, CSS:** Az HTML (Hypertext Markup Language) és a CSS (Cascading Style Sheets) az alapvető nyelvek a weboldalak szerkezetének és megjelenésének megtervezéséhez és implementálásához.
  - **Figma:** Ez egy online tervezőeszköz, amely lehetővé teszi webes és mobilalkalmazások tervezését, prototípusok készítését, valamint a tervezési folyamatok megosztását és együttműködését.

- A fejlesztési folyamat során a mindketten folyamatos hatékony kommunikáció folyt és együttműködésünk kiemelt jelentőséggel bírt.
- Fontos lépésnek találtuk az elkészült kódok rendszeres tesztelése és validálását annak érdekében, hogy biztosítsuk a weboldal megfelelő működését és stabilitását.
- A fejlesztés és implementáció során alkalmazott legújabb technológiai megoldások és fejlesztési módszertanok hozzájárultak a projekt hatékonyságához és minőségéhez.
- A fejlesztés során a GitHub-ot vettük igénybe, hogy minden fájl, kód mind a kettőnk számára elérhető legyen hiszen ezzel tudtuk garantálni, hogy a fejlesztés hatékony és zökkenőmentes legyen.

(<https://github.com/TudosLevente/ProjectWork>)

## Projektmenedzsment:

- ❖ *Budai Bence* – Front end felelős
- ❖ *Tüdős Levente* – Projektmenedzser, Back end felelős

### 1. Erőforrások hatékony kihasználása:

A projekt megosztása elég egyértelmű volt mivel mindketten tudtuk, hogy ki miben az erősebb.

### 2. Célok és felelősségi területek meghatározása:

A recept weboldal projektjének sikeres menedzselése és a határidő betartása érdekében kijelöltük a célokat és meghatároztuk a felelősségi területeket.

### 3. Folyamatos felügyelet a célkitűzések érdekében:

A projekt elkészítése folyamatos felügyelet alatt állt, hiszen egymást felügyeltük a projekt céljainak elérése és ütemezettség betartása érdekében.

### 4. Hatékony kommunikáció terv:

Kiemelkedő jelentőségű volt a kommunikációs terv kialakítása és végrehajtása, hogy hatékonyan tudjunk együttműködni és információkat cserélni. Erre szolgált a Messenger, Discord, Microsoft Teams.

### 5. Tapasztalatszerzés és felelősségvállalás a csapatban:

Mind a ketten sok tapasztalatot szereztünk hiszen megtanultunk egy kisebb csapatban fejleszteni és egymásra támaszkodni, felelősséget vállalni.

6. Proaktív problémamegoldás:

Amikor felmerült egy probléma vagy kihívás, proaktívan reagáltunk rá, és azonnal megpróbáltuk megtalálni a megoldást. Ez lehetővé tette számunkra, hogy gyorsan reagáljunk az esetleges akadályokra, és biztosítsuk a projekt zavartalan haladását.

7. Rugalmas tervezés és alkalmazkodás:

Amikor szükség volt változtatásokra vagy módosításokra a tervekben, rugalmasan alkalmazkodtunk és újraértékeljük a prioritásokat. Ez lehetővé tette számunkra, hogy rugalmasak legyünk az új információk és körülmények kezelésében, és sikeresen kezeljük az esetleges változásokat.

8. Teljesítményértékelés és tanulás:

Rendszeresen értékeltük a csapat teljesítményét és a projekt haladását, hogy azonosítsuk a sikereket és a fejlesztési területeket. Ez lehetővé tette számunkra, hogy tanuljunk az elért eredményekből és a tapasztalatainkból, és javítsuk a további projektekben való teljesítményünket.

9. Szoros együttműködés és támogatás:

Egymást támogattuk és segítettük a projekt teljes ideje alatt, és szorosan együttműködtünk azért, hogy sikeresen elérjük a kitűzött célokat. Ez a csapatmunka és egymásra támaszkodás kulcsfontosságú volt a projekt sikeréhez.

## Záró megjegyzések és köszönetnyilvánítás:

Az alábbiakban köszönetünket fejezzük ki Szilasi István osztályfőnökünknek és Nagy Ferenc tanárúrnak:

Kedves Szilasi István és Nagy Ferenc!

Teljes lelkünkben szeretnénk megköszönni önöknek a segítséget és támogatást, amit a projektünk során nyújtottatok. Szilasi István tanárúr, hálásak vagyunk önnek a back end kialakításában való segítségért és a tesztelésekben való segítségért. Nagy Ferenc tanárúr, köszönjük, hogy a front end, a dizájn és a tiszta kód elvei alapján való kódolásban volt nagy segítségünkre.

Az önök által nyújtott mentorálás és tudásanyag megosztásának köszönhetően sikerült hatékonyan és eredményesen dolgoztunk a projekten. Ezáltal nemcsak technikai ismereteink fejlődtek, hanem megértettük azt a fontosságot is, hogy egy csapatban való összefogás és együttműködés milyen kiemelkedő jelentőségű egy sikeres projekt megvalósításában.

Üdvözlettel,  
Budai Bence,  
Tüdős Levente



## A projekt munka telepítése:

Szükséges alkalmazások:

- **Visual Studio Code (VS Code)**
- **Node.js**
- **XAMPP**
- **MySQL Workbench (ajánlott)**

## Projektmunka telepítése:

Az első elindítása során telepítenünk kell a weblapot. A VS Code megnyitása után hozzunk létre egy új terminált ezek után írjuk bele a következő parancsot(kat):

```
git clone https://github.com/TudosLevente/ProjectWork.git
```

Ez után várjuk meg míg a klónozás befejeződik ezután kattintunk jobb egér gombbal a Js mappára:

```
cd ProjectWork
```

Ezzel megnyitjuk a mappát, ahova a le lett klónozva a projektmunka. Innen már csak kettő parancsot kell beírni:

```
cd js  
npm start
```

Az első parancs megnyitja a js mappát, a második pedig elindítja a szerveret, amin keresztül el lehet érni a weboldalt lokálisan.

```
A szerver fut | http://localhost:8000
```

Ha ezt az üzenetet kaptuk akkor már kész is van a szerver és minden fut, aminek kéne. Már csak arra kell figyelni, hogy a XAMPP alkalmazásban ellegyen indítva a MySQL szolgáltatás.

## Az adatbázis felépítése:

Az adatbázis .sql fájlt meglehet találni a Database mappában FoodHub.sql néven ez a fájl teljesen MySQL Workbench kompatibilis így ott egy kattintással le lehet futtatni.

## A weboldal megnyitása:

Ha már az sql-t is sikeresen lefuttattuk akkor már csak egy teendőnk van, mégpedig megnyitni a weboldalt a <http://localhost:8000/> címen.

## Weblapok:

### Főoldal:

A főoldalon található navigációs menü különféle fontos funkciókat kínál a felhasználók számára:

#### 1. Keresés mező:

Ezen a területen a felhasználók keresést végezhetnek az adatbázisban található receptek között. Egyszerűen beírhatják a keresett recept nevét vagy összetevőit, majd a rendszer megjeleníti a releváns találatokat.

#### 2. Bejelentkezés:

Ez a menüpont lehetővé teszi a felhasználók számára, hogy bejelentkezzenek vagy regisztráljanak a weboldalra. Ha már van fiókjuk, bejelentkezhetnek, ha pedig még nincs, akkor regisztrálhatnak az oldalon.

#### 3. Lenyíló menü elemek:

Itt a felhasználók választhatnak kategóriák szerinti böngészésre. Amikor egy kategóriát választanak, a rendszer átirányítja őket egy olyan oldalra, ahol az adott kategóriába tartozó receptek lesznek szűrve.

A főoldalon továbbá sok recept ajánlás is található, valamint lehetőség van a hírlevélre való feliratkozásra. Emellett a felhasználók visszajelzést küldhetnek a weboldal fejlesztőinek, és elfogadhatják az általános feltételeket, beleértve a sütik elfogadását is.

Bármely recept képére kattintva a felhasználók eljuthatnak a recept oldalára, ahol részletes információkat találnak a receptről, beleértve az összetevőket és az elkészítési módokat is. Ez lehetővé teszi számukra, hogy könnyen hozzáférjenek a kívánt receptekhez, és megnézzék azok részletes leírását.

## Bejelentkezés:

A bejelentkezés weblapon a felhasználók lehetőséget kapnak a weboldalra történő bejelentkezésre. Ha a felhasználó már regisztrált korábban, akkor megadhatja az e-mail címét, valamint a jelszavát az "Bejelentkezés" űrlapon keresztül.

Ha a felhasználó bejelentkezett, és csak a recepteket szeretné megtekinteni a kezdőoldalon, a navigációs sávban a következő lehetőségek jelennek meg:

- Receptek megtekintése:

Ez a menüpont egy link, amelyre kattintva a felhasználó át lesz irányítva a receptek oldalra, ahol böngészhet és keresés után recepteket találhat.

- Feliratkozás a hírlevélre:  
Ez a lehetőség lehetőséget biztosít a felhasználónak arra, hogy feliratkozzon a weboldal hírlevelére. Így értesítéseket kap a legfrissebb receptekről, újdonságokról és egyéb fontos információkról.
- Visszajelzés űrlap:  
Ez egy olyan űrlap, amely lehetővé teszi a felhasználók számára, hogy írjanak visszajelzést, észrevételeket vagy javaslatokat. Az űrlap tartalmazza a név mezőt, az e-mail cím mezőt és egy nagyobb szövegdobozt a visszajelzés leírásához.

Amennyiben a felhasználó még nem regisztrált a weboldalon, a "Még nem regisztráltál? Katt ide..." felíratra kattintva átirányítja őt a regisztrációs oldalra. Itt a felhasználók új fiókot hozhatnak létre, kitöltve az űrlapot a szükséges adatokkal.

Ezek a bejelentkezési és regisztrációs funkciók lehetővé teszik a felhasználók számára, hogy könnyen hozzáférjenek a weboldalhoz és használják annak szolgáltatásait. A bejelentkezés után a felhasználók teljes körű hozzáférést kapnak a profiljukhoz és az egyéb funkciókhoz, amelyeket a weboldal kínál.

### Regisztráció:

A regisztrációs weblapon a felhasználók lehetőséget kapnak a weboldalra történő regisztrációra, hogy később használhassák annak funkcióit, mint például receptek feltöltése és kedvencekhez való hozzáadás funkciókat.

Amikor egy felhasználó először látogatja a weboldalt, és regisztrálni szeretne, a regisztrációs oldalra irányítják. Itt a felhasználók kitöltik az űrlapot, amely tartalmazza a szükséges információkat, mint például a felhasználónév, jelszó, e-mail cím. Miután a felhasználó kitöltötte az űrlapot és elküldte az adatokat, a rendszer regisztrálja és létrehozza a felhasználói fiókot.

Amennyiben a felhasználó már rendelkezik fiókkal, és csak be szeretne jelentkezni, a "Már van fiókod? Jelentkezz be." felíratra kattintva átirányítják őt a bejelentkezési oldalra. Itt a felhasználók megadják a már regisztrált e-mail címüket, valamint a hozzájuk tartozó jelszót, majd bejelentkeznek a rendszerbe.

Ezek a regisztrációs és bejelentkezési funkciók lehetővé teszik a felhasználók számára, hogy könnyen hozzáférjenek és használják a weboldal szolgáltatásait, mint például receptek feltöltése és kedvencekhez való hozzáadás. A regisztráció és bejelentkezés folyamata egyszerű és intuitív, ami javítja a felhasználói élményt és segíti a felhasználókat a weboldal használatában.

## Recept kategória:

A recept kategória weblapra való eljutás lehetősége minden egyes oldalról elérhető a navigációs menü segítségével. Amikor a felhasználó bármelyik oldalon böngészik, a navigációs menü tartalmaz egy linket, menüpontot, amely "Kategóriák" elnevezéssel rendelkezik.

Amikor a felhasználó rákattint erre a linkre, menüpontra, átirányítást kap a recept kategória weblapra. Itt a recepteket különböző kategóriákba rendezhetik, például „Levesek”, „Főételek”, „Desszertek”, „Vegetáriánus” stb. Minden kategóriához egy kis ikon vagy kép is társítható, ami segíthet a felhasználóknak könnyen azonosítani és választani a különböző típusú receptek között.

A kategóriánkénti szűrés lehetősége a recept kategória weblapon is elérhetővé válik. Amikor a felhasználók egy adott kategóriát választana, akkor rendszer átirányítja őket egy olyan oldalra, ahol csak az adott kategóriába tartozó receptek jelennek meg.

Ezen az oldalon a recepteket kártyák formájában jelennek meg, ahol minden egyes recept alatt látható a recept neve, valamint egy kis kép a recept szemléltetésére.

A kiválasztott kategóriába tartozó receptek megjelenítése lehetőséget ad a felhasználóknak arra, hogy könnyen és gyorsan megtalálják az érdeklődésüknek megfelelő recepteket, anélkül, hogy át kelljen nézniük az összes rendelkezésre álló lehetőséget.

Ez a funkció javítja a felhasználói élményt és hatékonyabbá teszi a receptek böngészését, mivel lehetővé teszi a felhasználók számára, hogy azonnal megtalálják a kívánt kategóriába tartozó recepteket, anélkül, hogy további szűréseket kellene alkalmazniuk.

## Profil:

A profil oldal a felhasználók személyes központja a weboldalon, ahol számos információt és funkcionalitást találnak.

- Itt találhatóak meg a felhasználó által megadott adatok, melyek lehetnek a név, az e-mail cím. Emellett a profil oldalon elérhetőek a felhasználó által feltöltött receptek.
- Az elmentett receptek funkció lehetővé teszi a felhasználók számára, hogy kedvenc receptjeiket könnyen megtalálják és újra elkészítsék.
- Fontos kiemelni, hogy a felhasználói adatok és az interakciók biztonságát nagy figyelemmel kezeltük, és megfelelő adatvédelmi intézkedéseket alkalmazunk. A felhasználók biztonságának és védelme érdeken kulcsfontosságú a weboldal hitelességének és megbízhatóságának megőrzése érdekében.

## Recept feltöltés:

A recept feltöltés weblapon a regisztrált felhasználó tud új receptet feltölteni, ahol a képet mentsük egy mappába, ahol átnevezve tároljuk.

- Bejelentkezés és hozzáférés a feltöltési felülethez:  
Ha felhasználó bejelentkezik a weboldalra, akkor hozzáférhet a recept feltöltési felületéhez. Ez a felület lehetőséget biztosít a felhasználónak a recept nevének, leírásának, összetevőinek és elkészítési módjának megadására.
- Kép feltöltése:  
A felhasználó feltöltheti a recepthez tartozó képet a számítógépéről vagy egy másik eszközről. Amikor a kép kiválasztásra kerül, a weboldal egy mappába menti el, hogy később könnyen hozzáférhessen. Az általa feltöltött képet átnevezi, hogy egyértelműen azonosítható legyen a recepttel.
- Recept adatainak megadása:  
A felhasználó kitölti a recept adatait, beleértve a recept nevét, leírását, összetevőit és elkészítési módját. Ezen információk segítségével más felhasználók könnyen megtalálhatják és elkészíthetik a receptet.
- Mentés és közzététel:  
Miután a felhasználó kitöltötte az összes szükséges mezőt és feltöltötte a képet, elmenti és közzéteszi a receptet a weboldalon. Ezután más felhasználók láthatják és megoszthatják a receptet.

## Adminisztrátori felület:

Az adminisztrátori felület kizárólag az adminisztrátorok számára elérhető, akiket egy speciális jogosultság meghatároz az adatbázisban. Az adminisztrátoroknak számos fontos feladatot kell ellátniuk az admin felületen keresztül:

### 1. Feltöltött receptek kezelése:

Az adminisztrátoroknak lehetőségük van dönteni a felhasználók által feltöltött receptek sorsáról: jóváhagyhatják vagy törölhetik azokat.

1.1. Ha egy receptet jóváhagynak, az megjelenik a főoldalon és a kategóriák szerinti szűrési oldalakon.

1.2. Ha egy receptet törölnek, az nem kerül fel a platformra.

Ezzel az adminisztrátorok által végzett ellenőrzés biztosítja, hogy a weblapon csak minőségi és releváns tartalom legyen elérhető.

## 2. Hozzávalók kezelése:

Az adminisztrátorok kezelhetik és frissíthetik a hozzávalók listáját az adatbázisban. Ennek során új hozzávalókat adhatnak hozzá, meglévő hozzávalókat szerkeszthetnek, valamint felesleges vagy elavult hozzávalókat törölhetnek.

Ez biztosítja, hogy a weblapon mindig csak releváns és pontos hozzávalók legyenek elérhetőek a felhasználók számára.

## 3. Felhasználók kezelése:

Az adminisztrátorok kezelhetik és felügyelhetik a felhasználói fiókokat. Ez magában foglalja a felhasználói fiókok szerkesztését, deaktiválását vagy törlését, valamint új felhasználók regisztrálását.

Ezáltal lehet biztosítani a felhasználók fiókjainak biztonságát és hatékony kezelését.

## 4. Adminisztrátorok kezelése:

Az adminisztrátorok jogosultságot kapnak más adminisztrátorok kezelésére is. Ez magában foglalja új adminisztrátorok felvételét vagy meglévő adminisztrátorok eltávolítását.

Ezáltal biztosítva, hogy az adminisztrációs feladatokat hatékonyan lehessen végrehajtani a weboldalon, és hogy a rendszer mindig megfelelően működjön és felügyelt legyen.

Az adminisztrátori felület célja a teljes körű ellenőrzés és irányítás biztosítása a weboldal felett, lehetővé téve az adminisztrátorok számára, hogy hatékonyan kezeljék a tartalmat és a felhasználókat.

## Hibaoldal / 404 oldal:

A 404-es hibaoldal egy olyan weblap, amely akkor jelenik meg, amikor a felhasználó egy nem létező vagy elérhetetlen oldalra próbál navigálni. Ennek a hibaoldalnak a célja, hogy tájékoztassa a felhasználót arról, hogy az általa keresett tartalom nem található. A 404-es hibaoldal az itt felsorolt alábbi elemeket tartalmazza:

### 1. Hibaüzenet:

Világos és érthető üzenet, ami közli a felhasználóval, hogy a kért oldal nem található vagy nem elérhető.

### 2. Visszajelzés és irányítás:

Az oldal fontos eleme a link a főoldalra, amely lehetővé teszi a felhasználók számára, hogy könnyen visszatérjenek a webhely főoldalára.

### 3. Keresőmező:

A keresőmező lehetővé teszi a felhasználók számára, hogy keresési kulcsszavakat adjanak meg, így könnyebben megtalálják a kívánt tartalmat.

Ez a weboldal segít a felhasználóknak az oldalon való könnyű navigációban és az esetleges problémák gyors megoldásában. Emellett javítja az általános felhasználói élményt és hozzájárul az oldal hitelességéhez.

## Recept megtekintés:

A receptek megtekintése lehetővé teszi a felhasználók számára, hogy böngésszenek és megtekinthessék az oldalon található különböző recepteket. A receptek részletes információkat tartalmaznak, beleértve a képeket, hozzávalókat és az elkészítési útmutatókat. Ennek révén a felhasználók könnyen válogathatnak és inspirációt meríthetnek az ételkészítéshez.

- **Kedvencek mentése:**  
A kedvencek mentése funkció lehetővé teszi a felhasználók számára, hogy hozzáadjanak recepteket a kedvenceikhez. Ezáltal könnyen elérhetik és megtekinthetik ezeket a recepteket később, anélkül, hogy újra meg kellene keresniük őket az oldalon. Ez különösen hasznos lehet olyan receptek esetében, amelyeket a felhasználók később elkészítenének vagy újra meg szeretnének próbálni.
- **Kommentelés:**  
A kommentelés lehetőséget biztosít a felhasználóknak arra, hogy megosszák tapasztalataikat, véleményüket vagy kérdéseiket a receptekkel kapcsolatban. Ez interaktív módon segíti a felhasználókat abban, hogy kommunikáljanak egymással és az ételkészítés iránti szenvedélyüket megosszák. A kommentek segíthetnek más felhasználóknak a receptek jobb megértésében vagy továbbfejlesztésében, valamint közösségi élményt nyújtanak az oldalon.

Összességében ezek a funkcionalitások segítik a felhasználókat az oldalon való interaktív és hasznos tevékenységekben, és hozzájárulnak az ételkészítési élmény fejlesztéséhez és gazdagításához.

## Végpontok:

### Felhasználó:

#### /userRecipes/:id

- Végpont: /userRecipes/:id
- Metódus: GET
- Leírás:
  - Ez a végpont lekérdezi, hogy egy felhasználó hány receptet töltött fel, illetve azoknak mi az ID-ja, neve és képe.
- Paraméterek:
  - id: A felhasználó azonosítója vagy azonosítója, akinek a recepteit lekérdezzük.
- Hibakezelés:
  - Ha a megadott felhasználó azonosítóval nem található felhasználó, vagy ha nem rendelkezik feltöltött receptekkel, akkor üres választ kell visszaadni a státuszkód 404 Not Found mellett.

#### /user/:id

- Végpont: /user/:id
- Metódus: GET
- Leírás:
  - Ez a végpont lekéri egy regisztrált felhasználó összes adatát.
- Paraméterek:
  - :id: A felhasználó azonosítója vagy azonosítója, akinek az összes adatát lekérdezzük.
- Hibakezelés:
  - Ha a megadott felhasználó azonosítóval nem található felhasználó, üres választ kell visszaadni a státuszkód 404 Not Found mellett.

#### /regUser

- Végpont: /regUser
- Metódus: POST
- Leírás:
  - Ez a végpont regisztrál egy vadonatúj felhasználót a rendszerbe.
- Paraméterek:
  - A regisztrációhoz szükséges adatokat a kérés testében (body) kell elküldeni, általában JSON formátumban.
- Például:
  - Sikeres regisztráció:
    - Státuszkód: 200 OK
    - Tartalom: A regisztrált felhasználó adatai, vagy üzenet a sikeres regisztrációról.
  - Sikertelen regisztráció:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a regisztrációval kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.



## /updateUser

- Végpont: /updateUser
- Metódu: PUT
- Leírás:
  - Ez a végpont frissíti egy felhasználó feltöltött adatait.
- Paraméterek:
  - A frissítendő adatokat és a felhasználó azonosítóját (vagy azonosítóit) a kérés testében (body) kell elküldeni, általában JSON formátumban.
- Válasz:
  - Sikeres frissítés:
    - Státusz kód: 200 OK
    - Tartalom: Az frissített felhasználó adatai, vagy üzenet a sikeres frissítésről.
  - Sikertelen frissítés:
    - Státusz kód: 400 Bad Request
    - Tartalom: Hibaüzenetek a frissítéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /login

- Végpont: /login
- Metódu: POST
- Leírás:
  - Ez a végpont segít a felhasználó bejelentkezésében, hogy később recepteket tudjon feltölteni vagy azokat elmenteni.
- Paraméterek:
  - A bejelentkezéshez szükséges adatokat (például felhasználónév és jelszó) a kérés testében (body) kell elküldeni, általában JSON formátumban.
- Válasz:
  - Sikeres bejelentkezés:
    - Státusz kód: 200 OK
    - Tartalom: Az bejelentkezett felhasználó adatai, vagy üzenet a sikeres bejelentkezésről.
  - Sikertelen bejelentkezés:
    - Státusz kód: 401 Unauthorized vagy 400 Bad Request
    - Tartalom: Hibaüzenetek a bejelentkezéssel kapcsolatos problémákról.
  - Sikertelen bejelentkezés:
    - Státusz kód: 401 Unauthorized vagy 400 Bad Request
    - Tartalom: Hibaüzenetek a bejelentkezéssel kapcsolatos problémákról.

## Recept:

/recipe/:id

- Végpont: /recipe/:id
- Metódu: GET
- Leírás:
  - Ez a végpont segít lekérdezeni egy feltöltött recept adatait, hogy azt meg tudjuk jeleníteni.
- Paraméterek:
  - :id: A recept azonosítója vagy azonosítója, amelynek az adatait lekérdezzük.
- Válasz:
  - Sikeres lekérés:
    - Státusz kód: 200 OK
    - Tartalom: Az adott azonosítóhoz tartozó recept adatai.
  - Sikertelen lekérés:
    - Státusz kód: 404 Not Found vagy 400 Bad Request
    - Tartalom: Hibaüzenet, ha a megadott azonosítóval nem található recept.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

/uploadRecipe

- Végpont: /uploadRecipe
- Metódu: POST
- Leírás:
  - Ez a végpont segít feltölteni egy új receptet az adatbázisba.
- Paraméterek:
  - Az új recept adatait a kérés testében (body) kell elküldeni, általában JSON formátumban.
- Válasz:
  - Sikeres feltöltés:
    - Státusz kód: 201 Created
    - Tartalom: Az újonnan feltöltött recept azonosítója vagy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státusz kód: 400 Bad Request vagy 422 Unprocessable Entity
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /getComment/:id

- Végpont: /getComment/:id
- Metódus: GET
- Leírás:
  - Ez a végpont lekérdezi egy recepthez feltöltött kommenteket, beleértve a felhasználó nevét, a komment tartalmát és a feltöltés időpontját.
- Paraméterek:
  - :id: A recept azonosítója vagy azonosítója, amelyhez a kommenteket lekérjük.
- Válasz:
  - Sikeres lekérés:
    - Státuszkód: 200 OK
    - Tartalom: A megadott azonosítóhoz tartozó összes komment adatai.
  - Sikertelen lekérés:
    - Státuszkód: 404 Not Found vagy 400 Bad Request (függ attól, hogy a recept létezése hibakóddal vagy anélkül jelenik-e meg)
    - Tartalom: Hibaüzenet, ha a megadott azonosítóval nem található recept vagy hozzá tartozó komment.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /uploadComment

- Végpont: /uploadComment
- Metódus: POST
- Leírás:
  - Ez a végpont segít a felhasználóknak megírni véleményüket egy ételről.
- Paraméterek:
  - A komment adatait és az étel azonosítóját (vagy azonosítóit) a kérés testében (body) kell elküldeni, általában JSON formátumban.
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 201 Created
    - Tartalom: Az újonnan feltöltött komment azonosítója vagy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request vagy 422 Unprocessable Entity
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /getFavorites/:id

- Végpont: /getFavorites/:id
- Metódus: GET
- Leírás:
  - Ez a végpont lekérdezi egy regisztrált felhasználó által kedvencekhez hozzáadott recepteket.
- Paraméterek:
  - :id: A felhasználó azonosítója vagy azonosítója, akinek a kedvenceit lekérjük.
- Válasz:
  - Sikeres lekérés:
    - Státuszkód: 200 OK
    - Tartalom: A megadott felhasználó által kedvencekhez hozzáadott receptek azonosítói.
  - Sikertelen lekérés:
    - Státuszkód: 404 Not Found vagy 400 Bad Request
    - Tartalom: Hibaüzenet, ha a megadott azonosítóval nem található felhasználó vagy hozzá tartozó kedvencek.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /addFavorite

- Végpont: /addFavorite
- Metódus: POST
- Leírás:
  - Ez a végpont segít egy receptet egy már regisztrált és bejelentkezett felhasználónak a kedvencei közé adni.
- Paraméterek:
  - A recept azonosítóját és a felhasználó azonosítóját (vagy azonosítóit) a kérés testében (body) kell elküldeni, általában JSON formátumban.
- Válasz:
  - Sikeres hozzáadás:
    - Státuszkód: 200 OK vagy 201 Created
    - Tartalom: Az üzenet a sikeres kedvencekhez adásról.
  - Sikertelen hozzáadás:
    - Státuszkód: 400 Bad Request vagy 422 Unprocessable Entity
    - Tartalom: Hibaüzenetek a hozzáadással kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /removeFavorite

- Végpont: /removeFavorite
- Metódus: POST vagy DELETE
- Leírás:
  - Ez a végpont segít egy felhasználónak törölni egy esetlegesen véletlenül hozzáadott receptet a kedvencekhez.
- Paraméterek:
  - A recept azonosítóját és a felhasználó azonosítóját (vagy azonosítóit) a kérés testében (body) kell elküldeni, általában JSON formátumban.
- Válasz:
  - Sikeres törlés:
    - Státuszkód: 200 OK vagy 204 No Content
    - Tartalom: Az üzenet a sikeres kedvencek közül törlésről.
  - Sikertelen törlés:
    - Státuszkód: 400 Bad Request vagy 404 Not Found
    - Tartalom: Hibaüzenetek a törléssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /ingredients

- Végpont: /ingredients
- Metódus: GET
- Leírás:
  - Ez a végpont lekérdezi az összes hozzávalót az adatbázisban.
- Válasz:
  - Sikeres lekérés:
    - Státuszkód: 200 OK
    - Tartalom: Az összes hozzávaló listája.
  - Sikertelen lekérés:
    - Státuszkód: 500 Internal Server Error vagy 404 Not Found
    - Tartalom: Hibaüzenet az adatbázissal kapcsolatos problémák esetén.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre.

## /upload

- Végpont: /upload
- Metódus: POST
- Leírás:
  - Ez a végpont segít a felhasználónak feltölteni egy képet a receptjéről, azaz ételéről.
- Paraméterek:
  - A feltöltött képet és a hozzá tartozó adatokat (pl. recept azonosítója, felhasználó azonosítója stb.) a kérés testében (body) vagy a kép fejlécében kell elküldeni a multipart/form-data formátumban.
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK vagy 201 Created
    - Tartalom: Az újonnan feltöltött kép azonosítója vagy egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request vagy 422 Unprocessable Entity
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /notVerifiedRecipes

- Végpont: /notVerifiedRecipes
- Metódus: GET
- Leírás:
  - Ez a végpont segít az adminoknak lekérni a még nem hitelesített recepteket
- Válasz:
  - Sikeres lekérés:
    - Státuszkód: 200 OK
    - Tartalom: A még nem hitelesített recept címe, id-ja és képe
  - Sikertelen feltöltés:
    - Státuszkód: 404 Not Found
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /verifyRecipe

- Végpont: /verifyRecipe
- Metódus: PUT
- Leírás:
  - Ez a végpont segít az adminoknak hitelesíteni egy receptet
- Paraméterek:
  - A recept id-je
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
    - Tartalom: Egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /deleteRecipe

- Végpont: /deleteRecipe
- Metódus: DELETE
- Leírás:
  - Ez a végpont segít az adminoknak törölni egy esetlegesen kéretlen receptet.
- Paraméterek:
  - A recept id-je
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
    - Tartalom: Egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /giveAdminRole

- Végpont: /giveAdminRole
- Metódus: PUT
- Leírás:
  - Ez a végpont segít admin-t felvenni a rendszerbe
- Paraméterek:
  - Az admin email címe
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
    - Tartalom: Egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /deleteAdminRole

- Végpont: /deleteAdminRole
- Metódus: DELETE
- Leírás:
  - Ez a végpont segít egy esetlegesen véletlenül felvett admin-t törölni a rendszerből, vagy eltávolítani egy olyan személyt, aki már nem admin.
- Paraméterek:
  - Az admin email címe
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
    - Tartalom: Egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.



## /getAdmins

- Végpont: /getAdmins
- Metódus: GET
- Leírás:
  - Ez a végpont lekéri az összes jelenlegi Admin.
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /uploadIngredient

- Végpont: /uploadIngredient
- Metódus: POST
- Leírás:
  - Ez a végpont segít az adminoknak új hozzávalót felvinni a rendszerbe
- Paraméterek:
  - A hozzávaló neve, kategóriája.
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
    - Tartalom: Egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 409
    - Tartalom: Ez a hozzávaló már létezik.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /updateIngredient

- Végpont: /updateIngredient
- Metódus: PUT
- Leírás:
  - Ez a végpont segít az adminoknak egy esetlegesen elgépelte hozzávalót módosítani
- Paraméterek:
  - A hozzávaló neve
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
    - Tartalom: Egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## /removeIngredient

- Végpont: /removeIngredient
- Metódus: DELETE
- Leírás:
  - Ez a végpont segít az adminoknak egy esetlegesen rosszul felvitt hozzávalót törölni.
- Paraméterek:
  - Az hozzávaló id-je
- Válasz:
  - Sikeres feltöltés:
    - Státuszkód: 200 OK
    - Tartalom: Egy üzenet a sikeres feltöltésről.
  - Sikertelen feltöltés:
    - Státuszkód: 400 Bad Request
    - Tartalom: Hibaüzenetek a feltöltéssel kapcsolatos problémákról.
- Hibakezelés:
  - Az érvénytelen kérés vagy hibák esetén a megfelelő hibakódokkal és üzenetekkel kell válaszolni a kérésre. A feltöltött fájl típusának ellenőrzése és más hibás feltöltési paraméterek kezelése a fontos teendők közé tartozik.

## Hivatkozások:

- Cypress - [Cypress Dokumentáció](#)
- freeCodeCamp - [freeCodeCamp](#)
- MDN Web Docs - [MDN Web Docs](#)
- Node.js - [Node.js Dokumentáció](#)
- Stack Overflow - [Stack Overflow](#)
- W3Schools - [W3Schools](#)
- YouTube - [YouTube](#)
- Copilot képgenerátor - [Copilot](#)