



Eötvös Loránd Tudományegyetem

Informatikai Kar

Informatikatudományi Intézet

Programozáselmélet és Szoftvertechnológia Tanszék

Ételajánló webalkalmazás

Szerző:

Gál Rupert Bence

Programtervező informatikus BSc.

Témavezető:

Borsi Zsolt Richárd

Egyetemi tanársegéd

Budapest, 2024

Tartalomjegyzék

1.	Bevezetés	1
2.	Felhasználói dokumentáció.....	2
2.1.	A megoldott probléma rövid megfogalmazása	2
2.2.	Az alkalmazás telepítése és beüzemelése.....	2
2.2.1.	A program használata.....	2
2.2.2.	Futtatás.....	3
2.2.3.	Környezet.....	4
2.2.4.	Rendszerkövetelmények és támogatott platformok	4
2.3.	A program használata.....	5
2.3.1.	Nyelvváltoztatás	5
2.3.2.	Jogosultságok és főbb funkciók.....	5
2.3.3.	Információs üzenetek.....	6
2.4.	Adminisztrátor által hozzáfért felületek.....	6
2.4.1.	Entitások.....	7
2.4.2.	Adminisztrációk	9
2.5.	Felhasználó által elért felületek	10
2.5.1.	Jelszó módosítás.....	10
2.5.2.	Beállítások	11
2.6.	Bármely jogosultsággal elérhető oldalak	13
2.6.1.	Felhasználói fiók fül.....	13
3.	Fejlesztői dokumentáció	29
3.1.	Felhasznált módszerek részletes leírása	29
3.1.1.	Futtatási környezet.....	29
3.1.2.	Fejlesztői környezet.....	29
3.1.3.	Generálás.....	29

3.2.	Követelmény elemzés	30
3.2.1.	Felhasználók	30
3.2.2.	Felhasználói követelmények	30
•	Adminisztrátor képes:	30
3.2.3.	Funkcionális követelmények use case diagrammal szemlélítve.....	32
3.2.4.	Nem funkcionális követelmények	32
3.3.	A szoftver logikai és fizikai szerkezetének leírása	34
3.3.1.	Szerver és kliensoldali keretrendszer???	34
3.3.2.	Az alkalmazás felépítése.....	34
3.3.3.	Adatbázis	36
3.3.4.	Projekt struktúrája	46
3.4.	Tesztelés	47
3.4.1.	Tesztelési terv és tesztelési eredmények.....	47
4.	Összefoglalás és további fejlesztési lehetőségek	56
4.1.	Összefoglalás	56
4.2.	Továbbfejlesztési lehetőségek	57
5.	Irodalomjegyzék	58

1. Bevezetés

Egyik legfontosabb mégis időigényes és kihívásokkal teli feladat a mindennapi életben a főzés, főleg, ha változatos, finom és egészséges ételeket szeretnénk készíteni magunknak vagy a családunk számára. Napjaink felgyorsult világában, recept keresésre, étel kiválasztásra, bevásárlásra, vagy csak szimplán a főzésre időt keríteni egyre nagyobb kihívást jelent. Bár rengeteg információ áll rendelkezésre online szolgáltatóktól, ez a fajta hozzáférés a számtalan új recepthez, főzési ötlethez, tipphez már a pusztán mennyisége nehezzé teszi a tájékozódást közöttük, hogy megtaláljuk megfelelő számunkra. Az átláthatatlan receptadatbázisokban a felhasználók sokszor csak elvesznek a biztosított információ mennyiségtől és akár még a főzés idején is képes túlmutatni maga a keresési folyamat.

Továbbá bonyolítja a helyzetet, hogy a legtöbb elérhető recept gyűjtemény nem nyújt elég testreszabhatóságot. Az emberek nagyon változatos ízlésvilággal, eltérő diétával, allergiákkal és megszorításokkal rendelkeznek, emellett befolyásoló lehet még az otthon elérhető alapanyagaik, ezáltal a kivitelezhető ételeik és főzési választásuk. Például, egy laktóz érzékeny vagy egy vegán felhasználó számára jelentősen több időt és energiát vehet igénybe az ételek keresése és kiszűrése, hogy megegyezzen a követelményeikkel. Szintén egy gyakori gond lehet még, hogy nincs meg az összes alapanyag a recept elkészítéséhez, ami szükségessé teheti akár a hatalmasabb bevásárlást vagy a recept kihagyását.

A bevásárlólisták elkészítése, vagy a letisztult főzési instrukciók hiánya gyakran megakadályozza az embereket, a gyors és hatékony étel előkészítésben. A tradicionális receptes platformokról gyakran hiányoznak funkciók, amelyekkel a felhasználók könnyen számon tudnák tartani recepteket és követni a strukturált, következetes főzési folyamatokat, zökkenőmentessé téve elkészítést.

Ennek a szakdolgozatnak a célja kreálni és felépíteni egy applikációt, ami egyszerű, gyors és kényelmes megoldást nyújt a receptek keresésére, elmentésére vagy megosztására a felhasználók számára. Az alkalmazás fő célja, hogy felhasználóbarát felülettel hatékony eszköztárat biztosítson, amellyel fogyasztók könnyedén böngészhetnek receptet, gyorsan megtalálhatják az étkezési igényeiknek és alapanyagkészletüknek megfelelő recepteket, saját recept feltöltésre van lehetőség vagy akár össze is állíthatják receptes gyűjteményüket. A

fejlesztési folyamat során hangsúlyos, hogy felhasználók könnyen hozzáférjenek az ételekre vonatkozó alapanyag listához és lépésről lépésre követni tudják a főzés menetét.

A szakdolgozat részletesen bemutatja az alkalmazás fejlesztési folyamatát, a megvalósított funkciókat és azokat a technológiai megoldásokat, amelyek az intuitív és felhasználóbarát élmény biztosítása érdekében lettek integrálva. A dolgozat célja, hogy rávilágítson a mindennapi főzés során felmerülő kihívásokra és azok hatékony megoldására, amelyet ez az applikáció biztosít a felhasználók számára.

2. Felhasználói dokumentáció

2.1. A megoldott probléma rövid megfogalmazása

Az alkalmazás célja, hogy egyszerű, gyors és kényelmes megoldást biztosítson a felhasználóknak a receptek keresésére, mentésére és megosztására. A mindennapi főzés során sokszor problémát jelenthet, hogy a felhasználók nem találják azokat a recepteket, amelyek illenek az ízlésükhöz, az aktuális alapanyagaikhoz vagy az étkezés igényeikhez. Ezen kívül, a bevásárlólista és az étkezési lépések megszervezése is időigényes és kényelmetlen lehet, ha nincs megfelelő eszköz a receptek átlátható kezelésére.

Az alkalmazás megoldja ezeket a problémákat azáltal, hogy kategorizált receptböngészést, receptre szabott alapanyaglistát és lépésről lépésre következő főzési útmutatót biztosít. Ezen felül lehetővé teszi a felhasználók számára, hogy kedvenc receptjeiket elmentsék, és akár saját receptkönyveket összeállítsanak kedvenc receptjeikből. Az applikáció a felhasználók étkezési igényeinek megfelelően, intuitív módon segíti a receptek gyors és kényelmes keresését, miközben egy felhasználóbarát felületet biztosít főzéshez szükséges összes információval, egy helyen.

2.2. Az alkalmazás telepítése és beüzemelése

2.2.1. A program használata

Futtatáshoz és fejlesztéshez egyaránt szükséges a Java 17 vagy 21 LTS verzió, ami magába foglalja a Java SE Runtime 8-at, Java Virtual Machine-t is. Ezek mind elérhetőek a Java hivatalos oldalán: <https://www.java.com/en/download/manual.jsp>

Az alkalmazás futtatásához, illetve a csomagok könnyű behúzásához, letöltéséhez mindenképpen szükség van Node.js-re, ami esetében a legújabb LTS verzió kell. Ezt az alábbi linken lehet megtenni. <https://nodejs.org/en>

A fejlesztéshez ezen felül még szükség van Maven-re, ami az automatikus projekt felépítésért felelős. Itt található Apache Maven (<https://maven.apache.org/download.cgi>).

Futtatáshoz szükség van egy futtatási környezetre, általában a Java alkalmazásoknál a szimpla és egyszerű futtatáshoz az IntelliJ IDEA ajánlott, mint ahogy ebben az esetben is (<https://www.jetbrains.com/idea/download/>), persze más környezetek is mint a Spring Tools 4 is megfelelő VSCode vagy Eclipse extension formájában.

Amint az fentebb felsorolt programok feltelepítésre és beállításra kerültek, a projekt gyökérkönyvtárában az alábbi parancsot kell végrehajtani parancssorban, hogy a hiányzó könyvtárak letöltésre kerüljenek: 'npm install'

2.2.2. Futtatás

A futtatáshoz először is szükség van a fentebb említett lépések végrehajtására a megfelelő verzió számokat is figyelembe véve.

Következő lépés a projekt legújabb, legfrissebb verziójának letöltése, ezt a GitHub könyvtárból le kell tölteni vagy klónozni és ha van fejlesztői környezetünk, akkor a projektet betölteni.

A szerver futtatás a fejlesztői környezettől függetlenül akár beépített funkciókat használva akár a projekt gyökérkönyvtárában megadott parancssori parancsokkal is lehetséges. Parancssorban a következőket kell futtatni:

```
mvn clean install
```

```
mvn spring-boot:run
```

Ezek után már a definiált végpontok futnak a <https://localhost:8080/api> URL alatt. Frontend futtatásához azonban még szükséges ugyanúgy a gyökérkönyvtárból végrehajtani a következő parancsot:

```
npm start
```

Ennek segítségével a projekt Vue.js által szolgáltatott kidolgozott UI is elérhető a <https://localhost:9000> -es cím alatt.

2.2.3. Környezet

A program Java Runtime Environment-en és a Java Virtual Machine-en fut. A backendhez Spring Boot lett használva, ami a Spring-re építő MVC modellt alkalmazó Java keretrendszer.

A UI Vue.js -ben készült, ami egy Javascriptre épülő keretrendszer. HTML, CSS és Javascript nyelveken keresztül egy deklaratív, komponens alapú programozási modell, ami hatékony fejlesztést biztosít a bármiféle UI fejlesztésre legyen az szimpla vagy komplex.

2.2.4. Rendszerkövetelmények és támogatott platformok

A megfelelő futás érdekében elengedhetetlen, hogy a felsorolt programokat nehézség nélkül tudja párhuzamosan kezelni az eszköz. Minden operációs rendszeren lehetséges az alkalmazás futtatása, azzal a kitétel, hogy JVM telepítve van.

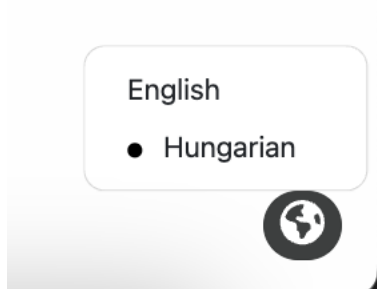
Minimum rendszerkövetelmények:

- Processzor: Kétféls processzor (például: Intel i3 vagy AMD megfelelője).
- Memória: 500MB
- Tárhely: 1GB

2.3. A program használata

2.3.1. Nyelvváltoztatás

Az oldalak jobb alsó sarkában található egy földgömb ikon, amely a nyelvválasztás funkcióját látja el. Az adatok alapértelmezetten angol nyelven jelennek meg, és ezek fordítására nincs lehetőség (angolul vannak tárolva), minden egyéb, kliensoldalon megjelenő elem szabadon változtatható a kiválasztott nyelvnek megfelelően.



ábra 1: Nyelvváltoztatás panel

2.3.2. Jogosultságok és főbb funkciók

Az alkalmazás funkcionalitását tekintve három jogosultságot volt szükséges definiálni, a megfelelő működést biztosítva. Ezek nem mások, mint az Adminisztrátor, Felhasználó és jogosultság nélküli Látogató.

A rendszerben háromféle jogosultsági szint mindegyikéhez eltérő hozzáférési körök tartoznak. Ezek célja, hogy a felhasználók csak az általuk engedélyezett funkciókhoz és adatokhoz férhessenek hozzá, növelve az adatok biztonságát és rendszer védelmét.

Az Adminisztrátornak mindenre kiható joga van, ezen keresztül bármilyen információt, adatot lát az applikációval kapcsolatban. Ez azt jelenti, hogy a bármilyen típusú adatot, relációt hozzá lehet adni, módosítani, törölni.

A Felhasználó jogosultsággal az alapvetőbb funkciók válnak elérhetővé, képesek vagyunk receptek hozzáadására, saját receptek módosítása, a már meglévő összes recepthez -saját receptek kivételével- értékelés írására, receptek mentésére és ezzel párhuzamosan saját receptkönyv kreálásra és összeállítására. Továbbá képes saját profiljára vonatkozó adatainak módosítására is.

Látogató jogosultsággal él minden nem regisztrált alkalmazás használó. Ezt a jogkört magába foglalja a regisztráció, más Felhasználók által hozzáadott receptek közötti böngészés a kereső

funkciót használva, receptkönyvek böngészése, megtekintése és a Felhasználó profilok oldalának megtekintése. Nincs lehetőség értékelés írására, se receptek, se receptkönyvek hozzáadására.

2.3.3. Információs üzenetek

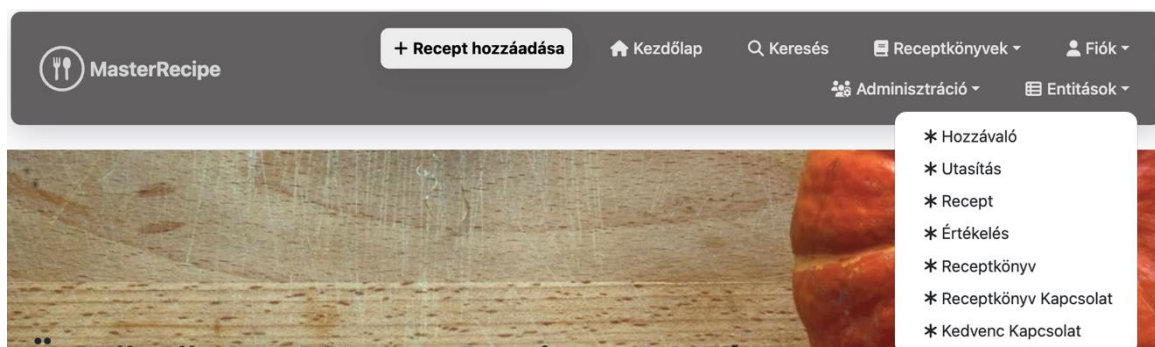
Az alkalmazás teljes használata során a felhasználót műveletek végrehajtását követően értesítő felugró üzenetek tájékoztatják az eredményről. Ezek az üzenetek három típusba sorolhatóak: információ, hiba és siker. A különböző típusok elsősorban megjelenésükben és az adott műveletre vonatkozó visszajelzés tartalmában térnek el egymástól.



ábra 2: Információs üzenet (Sikeres változat)

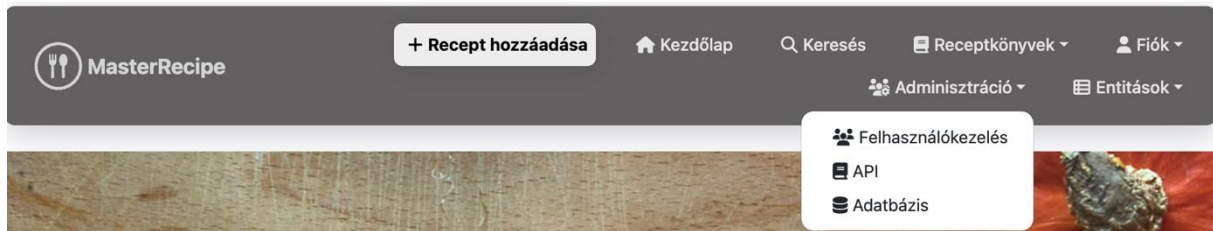
2.4. Adminisztrátor által hozzáfért felületek

Adminisztrátorként a navigációs sávon több új lehetőség is van a többi jogosultsághoz képest, entitásos és adminisztrációs fülek, ezek egy lenyíló listát nyitnak a hozzá tartozó lehetőségekkel. Az extra fülek csak a legmagasabb felhatalmazással rendelkezve lehet elérni. Entításokon belül megtalálható az alkalmazáson belül használt összes adattípushoz vezető hivatkozás.



ábra 3: Navigációs sáv entitások fül

Az másik fülön belül magas szintű adminisztrátori lehetőségek vannak, mint például a felhasználókezelés.



ábra 4: Navigációs sáv adminisztrációs fül

2.4.1. Entitások

Navigációs sávból elérhető entitások egységes kinézettel rendelkező lista oldalra vezetnek, ahol az adatbázisban tárolt összes adat, attól függően melyiket választottuk, megtalálható az adott entitásra vonatkozóan.

Receptek

[Lista frissítés](#) [+ új recept létrehozása](#)

Azonosító	Cím	Leírás	Hozzávalók neve	Ételek kategóriája	Étel típusa	Kép URL	Létrehozás dátuma	Felhasználó
1	Sweet and Sour Chicken	Classic Chinese dish made with crispy chicken in a tangy sweet and sour sauce.	Chicken,Cornstarch,Vegetable oil,Sugar,Vinegar,Soy sauce,Sesame seeds	Kínai	Előétel	http://localhost:8080/images/database/sweet-and-sour-chicken.jpg	Oct 18, 2024, 2:00 AM	10
2	Spaghetti Carbonara	Traditional Italian pasta dish made with eggs, cheese, pancetta, and pepper.	Spaghetti,Pancetta,Eggs,Parmesan	Olasz	Tészta	http://localhost:8080/images/database/spaghetti-carbonara.jpg	Oct 19, 2024, 2:00 AM	8
3	Hungarian Goulash	Hearty stew made with beef, potatoes, and plenty of paprika.	Beef,Paprika,Onion,Garlic,Potatoes,Carrots,Broth	Magyar	Ragu	http://localhost:8080/images/database/hungarian-goulash.jpg	Oct 20, 2024, 2:00 AM	9

ábra 5: Entitások listája (Hozzávalók oldala)

Lehetőség van új adat létrehozására egy következő erre készített oldalon, megadható minden adat az adattípusra vonatkozóan.

Hozzon létre, vagy módosítson Alapanyagot

Név

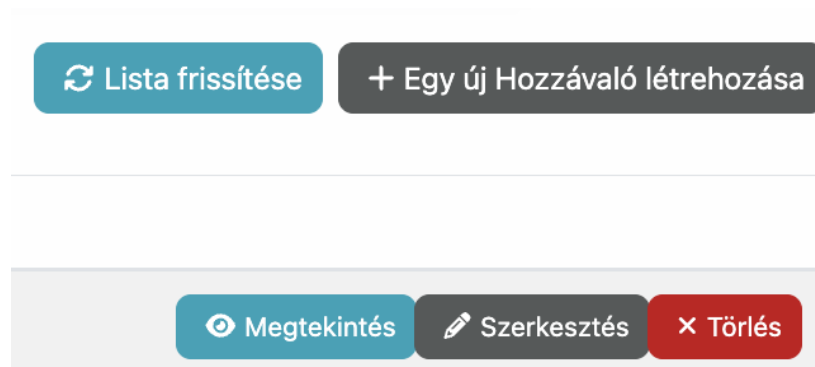
Mennyiség

Egység

Lépés

ábra 6: Új entitás (Új hozzávaló)

Van lehetőség ezen felül bármelyik meglévő adatnak a módosítására, ez tulajdonképpen megegyezik a létrehozás oldallal, annyi különbséggel, hogy az adatok kitöltődnek a mezőkben és csak a módosítást kell elvégezni. Egyedül a receptek esetében van egyedi oldal, erre később külön ki lesz térve.

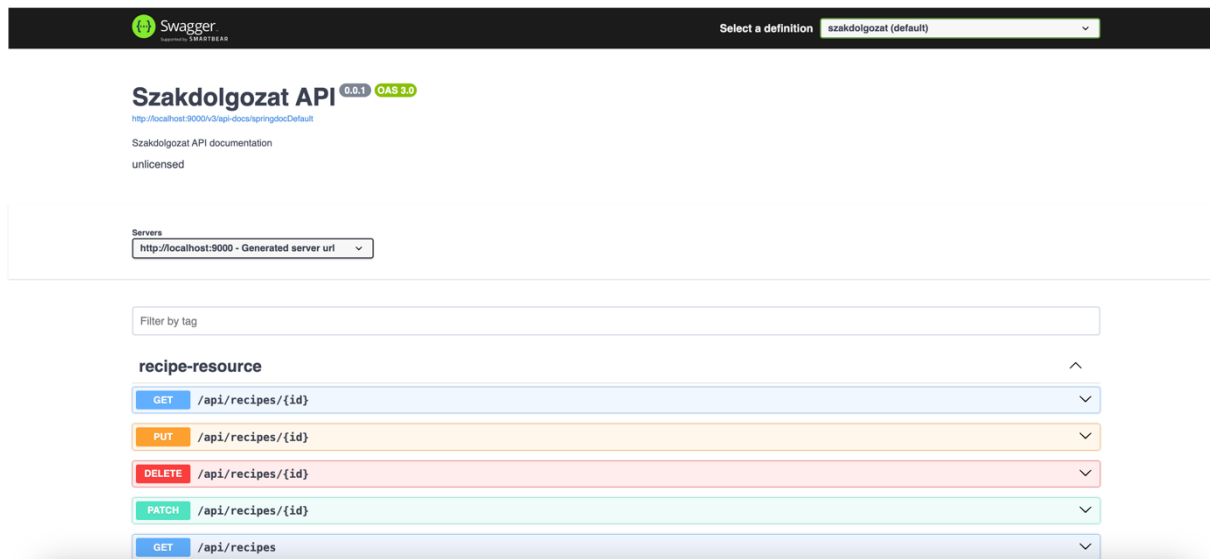


ábra 7: Entitások oldalon elérhető műveletek

Felső sor az magasabb szintű funkciókat jelölik, az alsó sorban pedig egy rekordra specifikusakat. A gombok magától értetődőek, 'megtekintés' az rekord áttekintéséhez, 'szerkesztés' a módosításhoz, 'törlés' pedig a törléshez van.

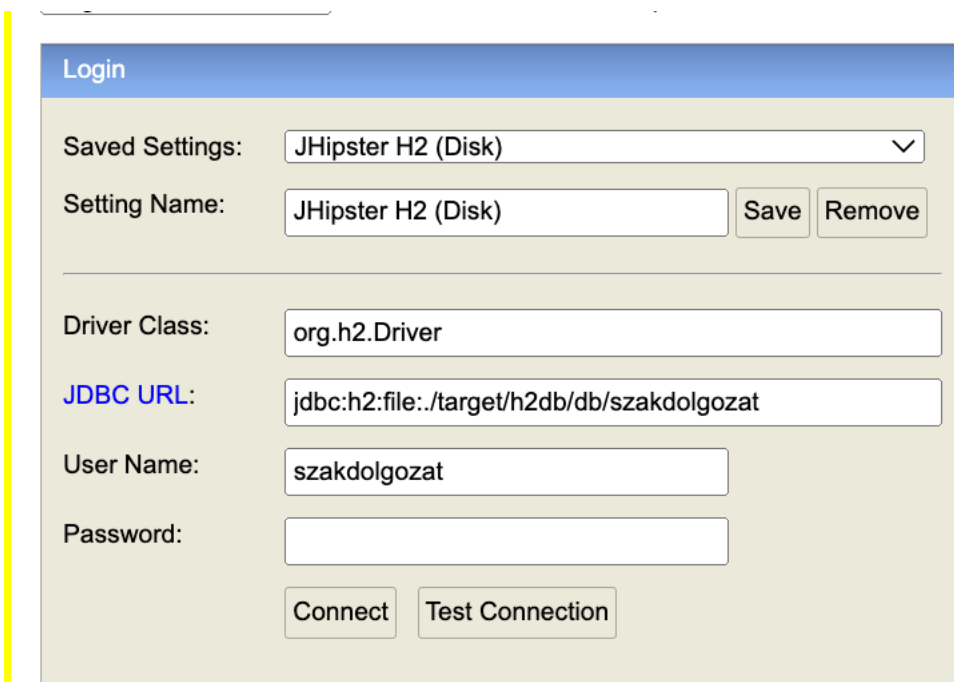
2.4.2. Adminisztrációk

Adminisztrációs listában található a felhasználókezelő, ami nagyban hasonlít a 2.4.1-ben bemutatott entitásokéval, viszont felhasználó specifikus funkciók is megtalálhatóak.



ábra 8: Swagger oldal (Elérhető végpontok)

Adminisztrációs navigációs fül tartalmaz továbbá még két másik opciót, az egyik az egész alkalmazáshoz tartozó összes API végpontot tartalmazó oldalra vezet. Itt ki lehet próbálni és használni saját megadott bemenetekkel ezeket. Valamint tartalmaz a fül még egy másik referenciát, amely segítségével az adatbázisunkhoz tudunk hozzáférni.



ábra 9: Adatbázis felület

2.4.2.1. Felhasználókezelő oldal

AZ oldal lényegében megegyezik az entitásoknál használt lista formával, azonban az alap megtekintés, szerkesztés és törlés funkciókon felül lehetőség van az adott felhasználó Profil oldalra is továbbmenni. Új felhasználó létrehozásának esetén nem a Látogatók számára elérhető regisztrációs oldalra visz, hanem egy részletesebb szerkesztőre, ahol bármelyik tulajdonságát meg lehet adni vagy manipulálni, beleértve a jogosultságokat is. Lehetőség van a felhasználók aktiválására, deaktiválására is.

Felhasználók

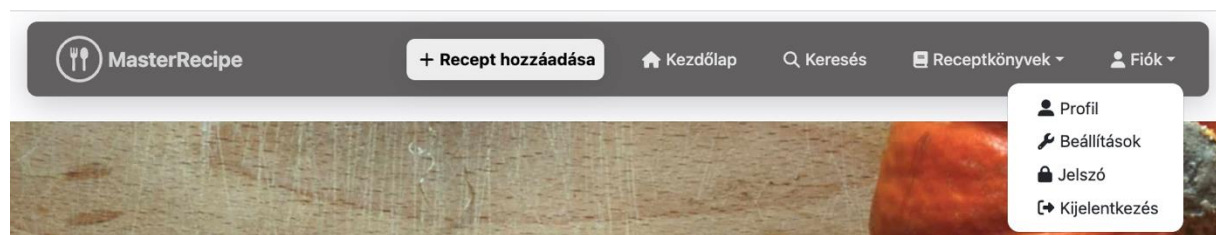
Lista frissítése + Új felhasználó

Azonosító	Felhasználónév	Email	Nyelv	Profil	Létrehozás dátuma	Módosította	Módosítás dátuma	
1	admin	admin@localhost	Aktív	en	ROLE_USER ROLE_ADMIN	Oct 18, 2024, 2:00 AM	system	1
								Profil megtekintése Megtekintés Szerkesztés Törés

ábra 10: Felhasználó kezelő

2.5. Felhasználó által elért felületek

Saját adataink módosítása(beállítások), jelszó csere, profilunk megtekintése és kijelentkezés a felhasználói fiók fül alatt található, mindegyik lehetőség egy-egy menüpontot jelent.



ábra 11: Navigációs sáv fiók fül

2.5.1. Jelszó módosítás

A bejelentkezett fiókhoz tartozó jelszó változtatására van lehetőség, saját jelszavunk megadásának fejében, és a mentés gomb megnyomásával lehet érvényesíteni. Ez a gomb csak a megfelelő bementek esetében válik aktívvá.

Jelszó [admin] számára

Jelenlegi jelszó

Új jelszó

Új jelszó megerősítése

Mentés

ábra 12: Jelszováltoztatás oldal

2.5.2. Beállítások

Saját fiókunk adatainak változtatására van lehetőség a beállítások oldalon. Felhasználónév kivételével minden adatot tudunk módosítani az oldalon, például vezeték-, keresztnév vagy az avatar.

Felhasználói beállítások [admin]

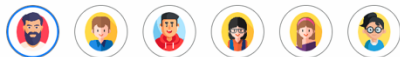
Keresztnév

Administrator

Vezetéknév

Administrator

Válassz ki az avatarodat!



Nyelv

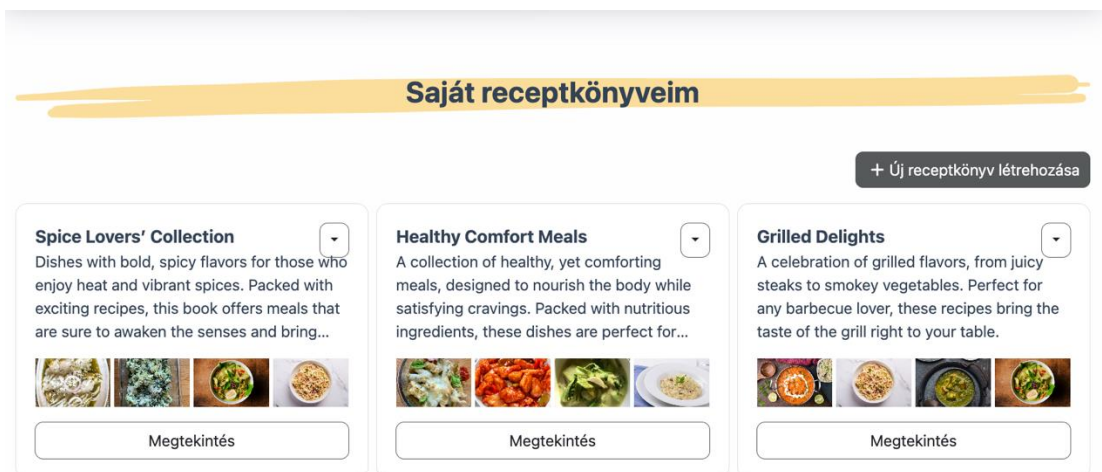
English

Mentés

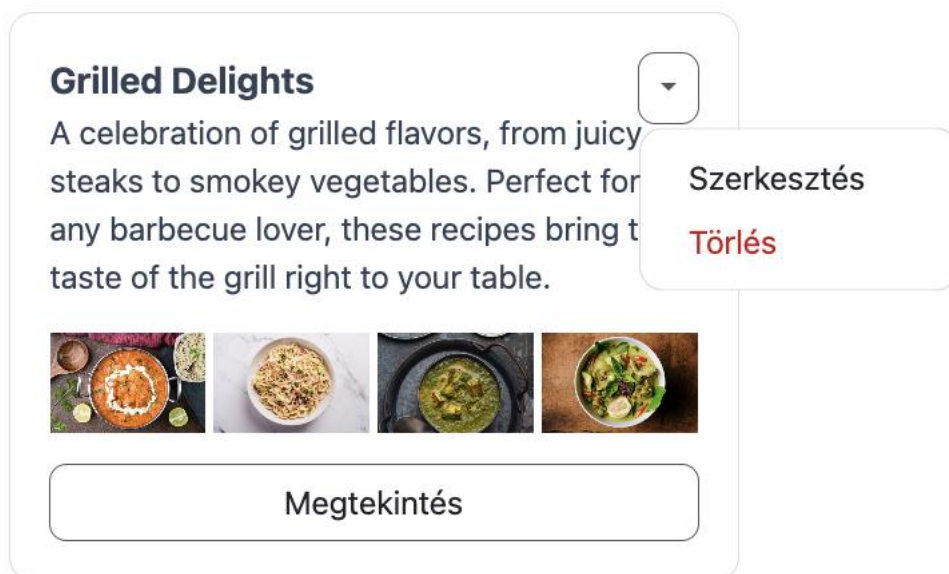
ábra 13: Beállítások oldal

2.5.2.1. Saját receptkönyvek oldal

A receptkönyvek fül alatt megtalálható opció, hogy a saját felhasználónkon keresztül létrehozott könyveket lehet megtekinteni. Ezen az oldalon megjelennek a publikus és privát receptkönyvek egyaránt, függetlenül a beállításainktól.



ábra 14: Saját Receptkönyvek oldal



ábra 15: Receptkönyv műveletek

Az oldal több műveletnek is helyet ad, főként a receptkönyvekre helyezve a hangsúlyt. A fenti kép mutatja a receptkönyvre specifikus opciókat, mint például a törlés vagy a szerkesztés. Ez csak tulajdonosként elérhető funkció. A szerkesztés gombbal egy dialógust tudunk megnyitni, amely segítségével a tulajdonságait, adatait tudjuk módosítani a kiválasztott gyűjteménynek. A dialógus megegyezik az új recept létrehozásánál alkalmazottal, annyi különbséggel, hogy az adatok előre ki lesznek ebben az esetben töltve és csak a változtatásokat kell elvégezni.

Receptkönyv szerkesztése

Név

Grilled Delights

Téma

Barbecue and Grilled Recipes

Leírás

A celebration of grilled flavors, from juicy steaks to smoky vegetables. Perfect for any barbecue lover, these recipes

Láthatóság

Privát

Mégse

Frissítés

ábra 16: Receptkönyv módosító dialógus

2.6. Bármely jogosultsággal elérhető oldalak

2.6.1. Felhasználói fiók fül

2.6.1.1. Regisztráció

Az oldal a felhasználó kreálást teszi lehetővé, a szükséges kitöltendő mezőket tartalmazza, amelyek magukba foglalják a felhasználónevet (ezzel fogunk tudni bejelentkezni a profilunkba), jelszót, vezeté- és keresztnévet, illetve az alkalmazásban több helyen is megjelenő avatárunkat kell kiválasztani az előre megadott lehetőségek közül.

Regisztráció

Felhasználónév

Új jelszó

Új jelszó megerősítése

Keresztnév

Vezetéknév

Válaszd ki az avatárodat!



ábra 17: Regisztrációs oldal

2.6.1.2. Bejelentkezés

Fiók menüpontból elérhető a bejelentkezés dialógus, ahol a felhasználónév és a jelszó kombináció segítségével lehetséges a belépés, ha még nem lenne fiókunk akkor van regisztrációra is opció. Az alkalmazásban jogosultsághoz kötött funkciók invokációja során szintén megjelenik a panel, kötelezve a felhasználót a bejelentkezésre, abban az esetben, ha ki lennénk jelentkezve.

Bejelentkezés

Felhasználónév

Az Ön felhasználóneve

Jelszó

A jelszavad

Bejelentkezés

Nincs még fiókja?
Regisztráljon itt.

ábra 18: Bejelentkezés panel

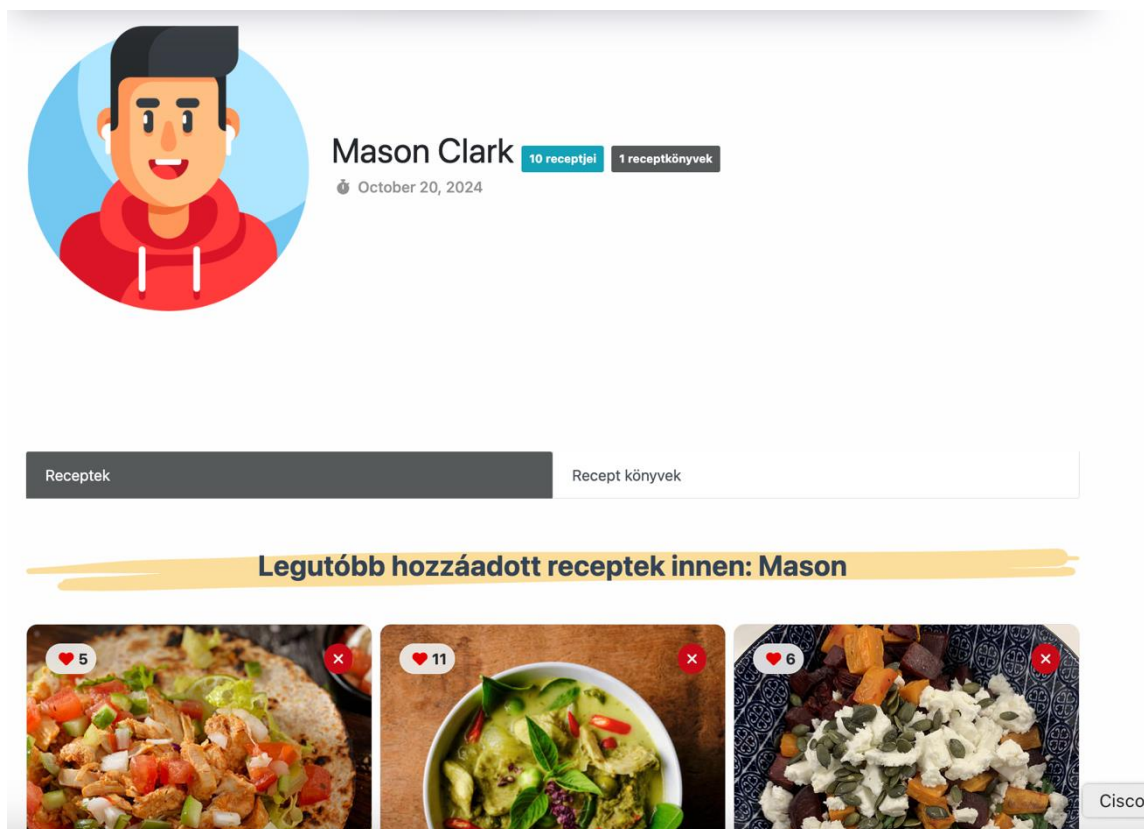
2.6.1.3. Profil oldal

A már létrehozott profilok közül bármelyik megtekinthető vendég vagy regisztrált felhasználó számára egyaránt. Az adott profilhoz tartozó adatok találhatóak itt, mint például az felhasználó aktiválásának kezdete, hogy hány receptet és receptkönyvet hozott létre, de helyet kapott profilhoz tartozó receptek és receptkönyvek listája is. Az utóbbiakhoz a közvetlenül felhasználói adatok után van lehetőség választani.

Receptek

Recept könyvek

ábra 19: Profil oldal lista műveletek



ábra 20: Profil oldal

2.6.1.4. Kereső

Az alkalmazáson belül a komplexebb oldalak közé tartozik, amely helyt ad egy szofisztikált keresőnek, lehetővé téve a szűrést receptekre több szempont alapján. Ezek a szempontok a recept neve, kategóriája, típusa és az elkészítéséhez szükséges hozzávalóit jelentik. Ezek a feltételek közül egyedül a címre való keresés szűkít, összes többi szempont bővíti a visszakapott eredmény halmazt. Abban az esetben, ha az összes kategóriát, típust és hozzávalót kiválasztanánk vagy semmit nem választanánk ki, akkor az egész adatbázisban szereplő összes receptet visszakapnánk.

Keresés

Válaszd ki, hogyan szeretnéd megtalálni a receptedet!

Recept címe

Írd be a recept címét...

Keresés

Étel típusok

Étel kategóriák

Hozzávalók

1. oldal

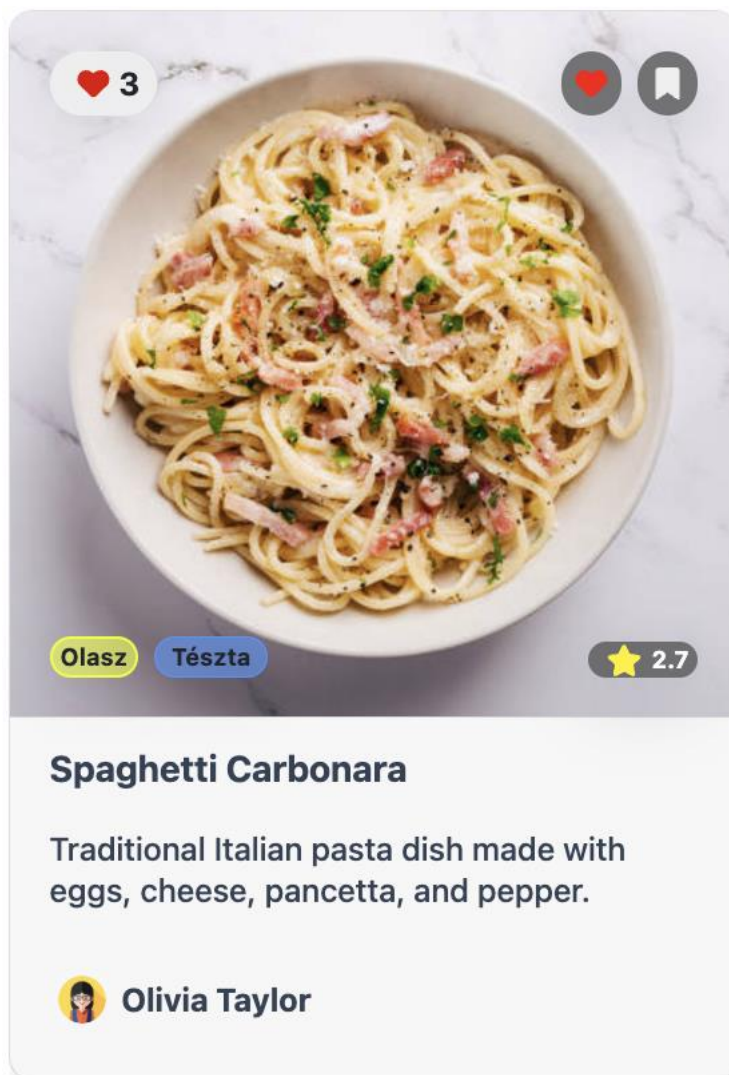
Létrehozás dátuma

90 találat(ok)

ábra 21: Keresési és rendezési kritériumok

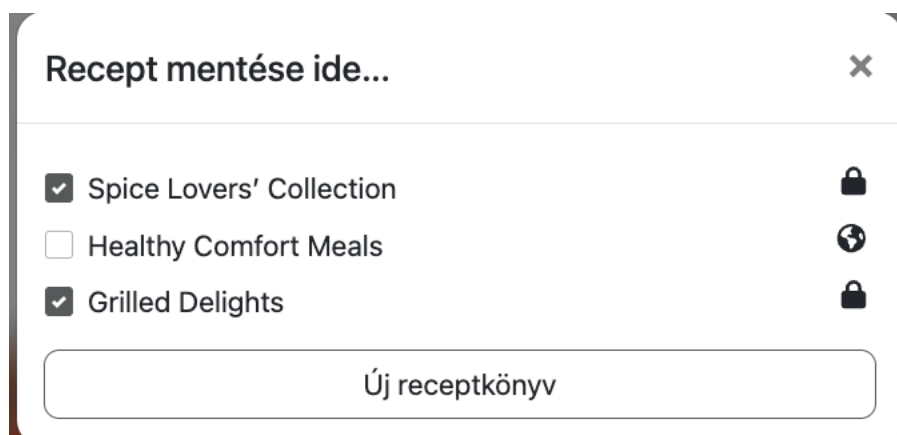
Továbbá van lehetőség ezen belül rendezni a listát szintén több tulajdonság alapján. Erre közvetlenül a kereső mezők alatti kiválasztható listával van lehetőség a rendezési tulajdonság megadására, mellette az ikon használatával pedig az irányt tudjuk állítani. Mezők közé tartozik a cím és a feltöltés dátuma, mindezek növekvő és csökkenő irányban is lehetségesek.

2.6.1.5. Recept kártya



ábra 22: Receptes kártya

A kereső oldalon megjelenő kártyák mindegyike egy receptet és annak adatait tartalmazza. A bal felső sarokban található szív ikon és szám a receptet kedvelő felhasználók számát jelenti. Jobb felső sarokban a két ikon közül a baloldali (Szív)-vel lehet kedvelni a receptet, a könyvjelző ikonnal pedig későbbre lehet elmenteni ezeket receptkönyveinkbe. Jobb alsó sarokban felhasználók által hagyott értékelések átlaga látszik csillaggal jelölve. Bal sarokba az étel kategóriáját és típusát találjuk, kép alatt pedig a címet, rövid leírását és készítőjét. Van lehetőség továbbá a szerző profiljára továbbmenni, ha érdekel minket a felhasználó által megosztott többi recept vagy receptkönyv.



ábra 23: Recept receptkönyvhöz adás panel

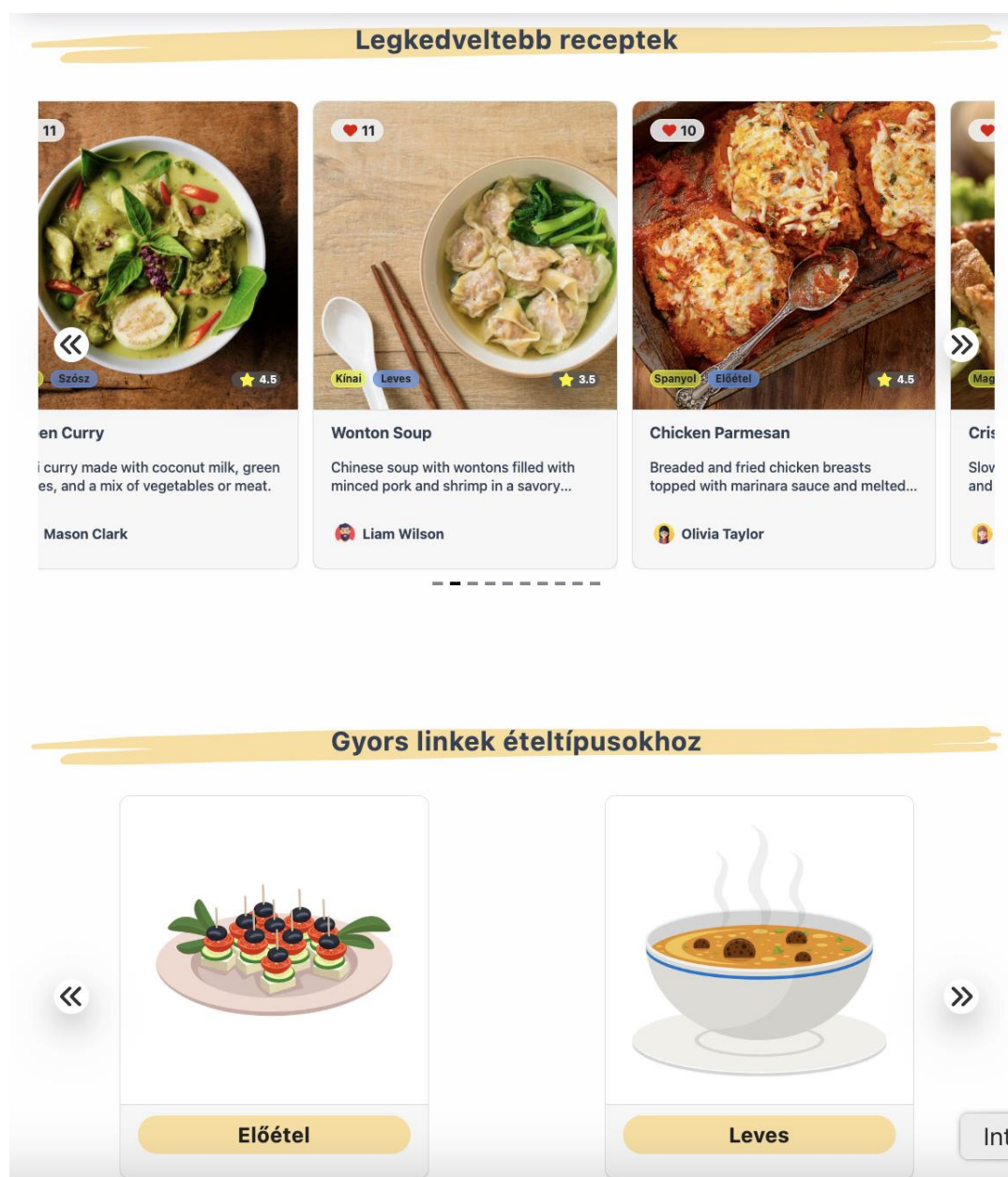
A dialógus segítségével tudjuk a kiválasztott receptet a kiválasztott receptkönyvbe rakni. Minden sor egy saját receptkönyvet mutat, jobb oldalon pedig a gyűjteményre vonatkozó láthatóság olvasható le az ikonok alapján (lakat – privát, földgömb - publikus). Új receptkönyv gomb az alábbi dialógust hozza fel, adatok kitöltése után létrehozható újabb receptkönyv.

A screenshot of a web application dialog box titled "Recept mentése ide...". The dialog has a close button (X) in the top right corner. It contains several input fields: "Név" with a placeholder "Receptkönyv neve...", "Téma" with a placeholder "Receptkönyv témája...", and "Leírás" with a placeholder "Receptkönyv leírása...". Below these is a dropdown menu for "Láthatóság" currently set to "Nyilvános". At the bottom of the dialog are two buttons: "Új receptkönyv" and "Mégse".

ábra 24: Új receptkönyv panel

2.6.1.6. Home

Fő oldal, egyben ide érkezik a felhasználó kezdéskor, de a navigációs sávon is megtalálható a ház ikon alatt. Mindig változó véletlenszerű sliderek vannak receptekkel, receptkönyvekkel és felhasználókkal feltöltve különböző szempontok alapján. Ezek főként az étel kategóriák, típusok, hozzáadás dátuma, kedvelések száma, magasan értékelt receptek és receptkönyv ajánlók, illetve néhány aktívabb profil is megtalálható. Az oldalon még van néhány az oldal folyékonyabbá tételére elhelyezett „hirdetést”, valamint egy gyorslinkeket tartalmazó carousel is előre megadott keresési paraméterrel.




ábra 25: Főoldal (részlet)

2.6.1.7. Recept oldal


Alkalmazás legfontosabb oldala, maguknak a recepteknek az oldala. Minden információ és adat, ami az adott recepthez köthető az megtalálható itt. Főbb jellemzése található az oldal felső részén, lejjebb haladva bal oldalon bevásárlólista-szerűen a szükséges alapanyagok mennyiséggel és mértékegységgel, mellette lépésről lépésre az elkészítési útmutató. Összes lépésnek megvan a sorrendje, rövid címe, ami a lépés rövid összefoglalását jelképezi, egy hosszabb leírása és az utasítás végrehajtásához szükséges időtartam. Az étel képe alatt jobboldalon megtalálható a recepthez tartozó két funkció, kedvelés vagy receptkönyvhöz hozzáadás – ezekre 2.6.1.5-ös pontban pontosabban kitértem. Továbbá jobb felső sarokba elérhető a módosítás gomb – ehhez csak a recept szerzőnek van hozzáférése.

← Receptek böngészése



Churros with Chocolate

Mexikói Desszert

 Liam Wilson

Mexican dessert with fried dough sticks dusted in sugar and served with chocolate.

Hozzávalók

150 **GRAMM** of Flour
250 **MILLILITER** of Water
30 **GRAMM** of Sugar
500 **MILLILITER** of Vegetable oil
100 **GRAMM** of Chocolate
50 **MILLILITER** of Heavy cream

Utasítások:

1. Mix flour, water, and sugar to form dough.
⌚ 15 minutes
2. Fry churros until golden brown.
⌚ 5 minutes
3. Serve churros with melted chocolate.
⌚ 10 minutes

Pre

ábra 26: Recept oldal (részlet)

Legalsó szekció pedig a hozzászólásoknak ad helyet, ahol saját magunk által létrehozott ételeknél ugyan nincs lehetőség, de mások receptjein tudunk értékelést, hozzászólást hagyni.

Erre biztosított 4 mező, amikben 1-5-ig tudjuk pontozni, egy rövid leírást, hosszú leírást és még képet is tudunk csatolni. Egyedüli kötelezően kitöltendő mező a pontozás, ezen felül a többi opcionális. Továbbá, mások véleménye is látható, viszont ez esztétikai okok miatt le van korlátozva kettőre.

Vélemények

★ 4.5

Írhatsz vagy szerkesztheted a véleményedet!

Értékel

☆

☆

☆

☆

☆

Cím

Írd be a vélemény címét


Leírás

Írd be a véleményed leírását

Kép


Choose file No file chosen

Értékelés

 Amelia Harris

★ ★ ★ ★ ★

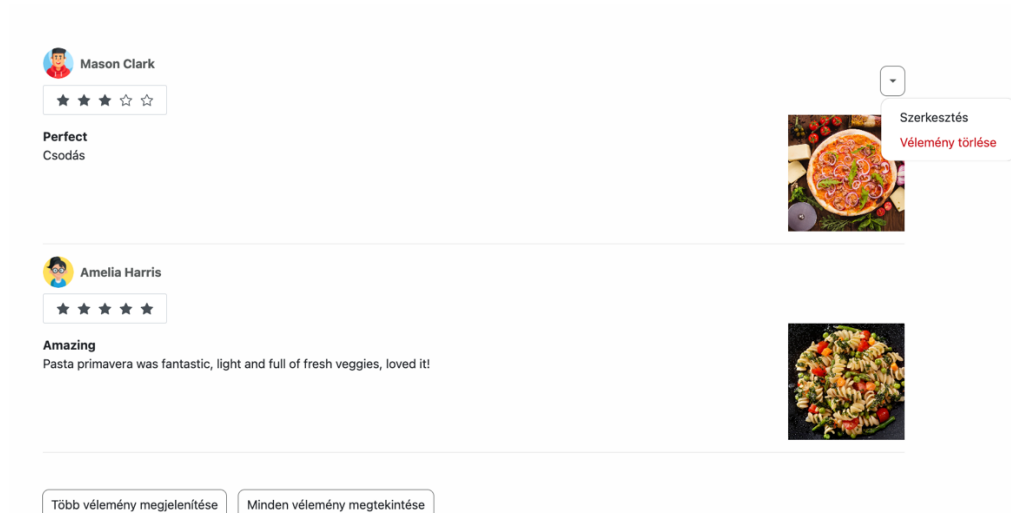
Amazing
Pasta primavera was fantastic, light and full of fresh veggies, loved it!



ábra 27: Hozzászólás szekció (Recept oldal részlet)

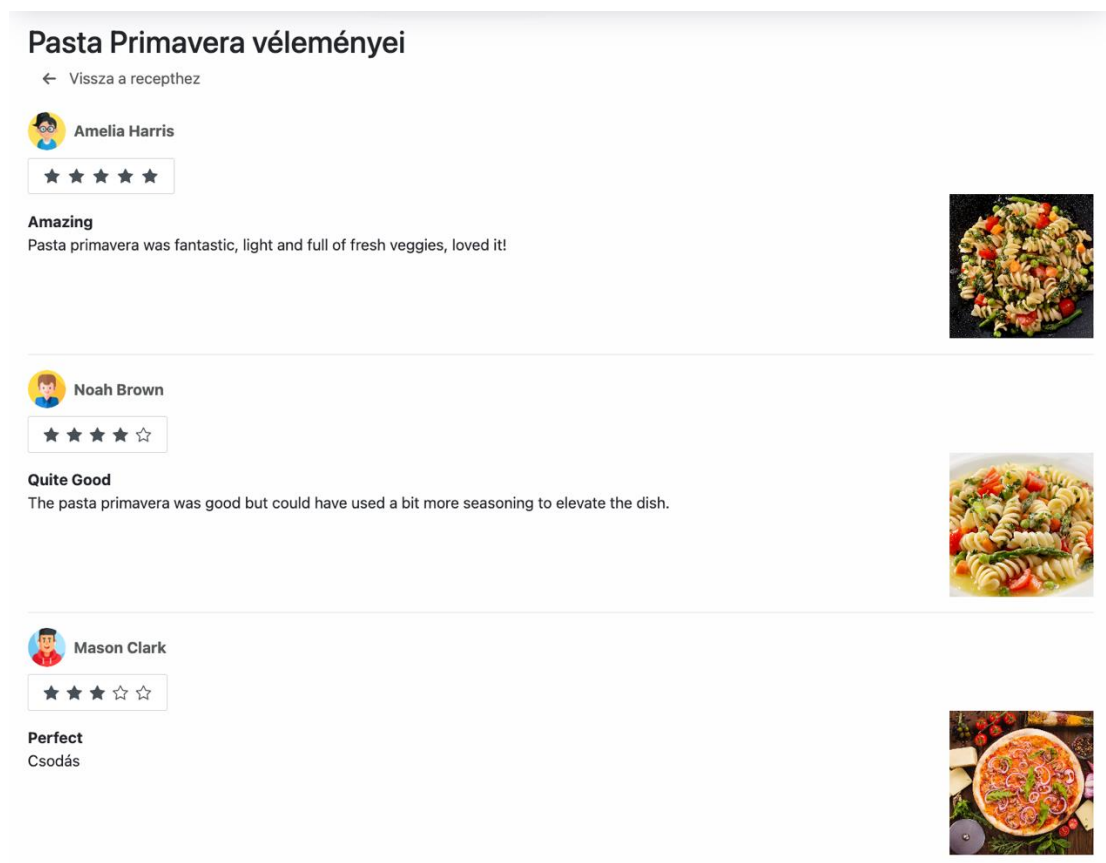
A hozzászólás szekcióban több műveletre is lehetőség van, ezek közül legalul található az említett, további hozzászólások megjelenítése vagy átirányítás az összes, ehhez recepthez tartozó hozzászólás oldalra. Van lehetőség még, abban az esetben, ha már van hozzászólásunk a recepthez, hogy módosítsuk vagy töröljük azt. A módosítás kiválasztása után újra meg fog jelenni a hozzászólás írására szolgáló űrlap, ahol kitöltésre kerül a jelenlegi adatokkal.

22



ábra 28: Hozzászólás szekció műveletek

2.6.1.7.1 Recepthez tartozó összes hozzászólás



ábra 29: Recepthez tartozó hozzászólás oldal

2.6.1.8. Új recept

Minden felhasználó számára elérhető a navigációs sávból az recept kreáló oldal. Minden recepthez szükséges hozzáadni egy címet az ételről, leírást, kategóriát, típust és egy képet az

elkészült ételről. Továbbá szükséges főzési instrukciót is adni a recepthez. Mindezek közül az étel kategóriája és típusa csak egy előre megadott listából lehet kiválasztani.



Új recept létrehozása

Cím

Leírás

Ételek kategóriája

Étel típusa

Kép URL

No file chosen Browse

+ Lépés hozzáadása

Mégse Mentés

ábra 30: Recept hozzáadás oldal

Dinamikusan lehetőség van akármennyi instrukciós lépés hozzáadására, ami egy gomb segítségével lehetséges és egy dialógust fog felhozni. A dialóguson a főbb jellemzőket kell megadni, ide tartozik a lépés neve, leírása, igénybevétel ideje és szintén dinamikusan lehetőség van a hozzávalók felsorolására. A hozzávaló nevét egy már meglévő listából lehet kiválasztani és mellé szükséges az adott mennyiség és mértékegység, amiben ez értendő. Mértékegységek egyszerűség kedvéért a 'gramm' a tömegre, 'milliliter' a folyadékokra és 'darab' az általánosabb alapanyagokhoz. Abban az esetben, ha nem található az alapanyag, amire szükségünk lenne, csak adminisztrátor képes újat hozzáadni.

Lépés hozzáadása

Cím

Adja meg a lépés címét

Idő (percekben)

Adja meg a szükséges időt

Leírás

Adja meg a lépés leírását

1. Hozzávaló
Mennyiség
Mértékegység

2. Hozzávaló
Mennyiség
Mértékegység

+ További hozzávaló hozzáadása

Hozzáadás

ábra 31: Új instrukció hozzáadás panel (Új recept oldal részlet)

2.6.1.9. Receptkönyv böngésző oldal

A kereső oldalhoz hasonlóan van a receptkönyvekre vonatkozó, kevésbé specifikus böngészési lehetőség, hogy a már létrehozott, publikus könyvek között is legyen lehetőség keresni. Az oldalon megjelenő receptkönyv kártyák segítségével megtekinthető azok szerzője vagy maga a könyv, valamint látható néhány kép a gyűjteményben tartalmazott receptekről.

25

Receptkönyvek

Ezek a nyilvános receptkönyvek

Receptkönyv keresése megadott kritériumok alapján

Receptkönyv címe

Receptkönyv leírása

Receptkönyv témája

Receptkönyv címkéje

Cím...

Leírás...

Téma...

Címke...


Keresés

1. Oldal


16 Receptkönyv

Global Comfort Foods

A collection of hearty dishes from around the world, perfect for cozy nights. Featuring warming flavors and rich textures, these recipes are designed to bring comfort and...




Megtekintés


 **Elijah Garcia**

Rich and Creamy Dishes

Recipes focusing on creamy, indulgent flavors for comforting meals. From velvety risottos to luscious curries, enjoy dishes that provide warmth and satisfaction in every...




Megtekintés


 **Ava Martinez**

Quick Weeknight Meals

Simple, quick meals for busy evenings that don't sacrifice flavor. These recipes ensure you enjoy hearty, satisfying dinners with minimal effort and maximum taste, perfect...



Megtekintés

 **Noah Brown**

ábra 32: Receptkönyv böngésző oldal

2.6.1.10. Receptkönyv oldal

Receptkönyvekhez tartozó információs oldal. A könyvhöz tartozó név, téma, leírás és címkék láthatóak. Jobb felső sarokban lehetőség van a szerkesztő számára, hogy mások számára elérhetővé vagy priváttá változtassa azokat. Közvetlenül mellette lévő szerkesztés gombbal a receptkönyvekre vonatkozó szerkesztő dialógust tudjuk felhozni és módosítani a gyűjteményt. Az oldal alsó része pedig a könyvben található receptek kártyái találhatóak, a tulajdonosnak van lehetősége receptek eltávolítására a gyűjteményből.

26

← Vissza

Ez a recept könyv privát ☐

Szerkesztés

Spice Lovers' Collection

Dishes with bold, spicy flavors for those who enjoy heat and vibrant spices. Packed with exciting recipes, this book offers meals that are sure to awaken the senses and bring bold tastes to your table.

#spicy

#boldflavors

#heat

#asianfusion

#chilies

#curry

8

×

MEXICAN

APPETIZER

Enchiladas Verdes

Tortillas filled with chicken and topped with green tomatillo sauce and cheese.

7

×

ITALIAN

NOODLE

Spinach and Chicken Pasta

Delicious pasta dish with spinach, chicken, and a creamy sauce.

11

×

THAI

SAUCE

Green Curry

Thai curry made with coconut milk, green chilies, and a mix of vegetables or meat.

ábra 33: Receptkönyv oldal

2.6.1.11. Not found

Az oldal nem létezik.

Vissza

Rossz URL cím megadása esetén az applikáció a 'Not Found' oldalra fog irányítani, a felhasználónak lehetősége van a navigációs sáv segítségével vagy a 'Vissza' gombbal valós oldalra elnavigálni.

2.6.1.12. Forbidden



Nincs jogosultságod az oldal eléréséhez.

Bejelentkezés

Vissza

A Forbidden vagy tiltott oldalra kapunk átirányítást, ha jogosultságunk nem elég magas, de mégis megpróbáljuk elérni azokat a felületeket, ahová szükség lenne rá. Egy jó példa erre bármelyik Adminisztrátori fülben megtalálható opció, akár felhasználóként akár vendégként.

3. Fejlesztői dokumentáció

3.1. Felhasznált módszerek részletes leírása

3.1.1. Futtatási környezet

A program futtatásához elengedhetetlen a Java Runtime Environment (JRE), amely a Java környezetet biztosítja, és a Java Virtual Machine, amely a Java kód végrehajtását teszi lehetővé. Ezen felül a Spring keretrendszer is fontos, mivel automatikusan behelyezi a szükséges függőségeket a példányosított osztályokba. A Maven pedig a függőségek kezelését látja el: automatikusan letölti és elérhetővé teszi a program által igényelt könyvtári elemeket.

3.1.2. Fejlesztői környezet

Platform:

- Lokális alapú szerver

Keretrendszerek:

- Spring Boot 3.3.4
- Vue.js: 3.5

Fejlesztői környezet:

- IntelliJ IDEA: 2024.1

Felhasznált nyelvek és eszközök:

- Java: 17.0.10+8-b1207.12
- JavaScript / TypeScript (Vue.js-hez)

3.1.3. Generálás

A projekt kezdeti fázisában a modern, ingyenesen elérhető és szabad forráskódú (JHi1) applikációgenerátor került felhasználásra. A legnagyobb előnyei közé tartozik a különböző keretrendszerek integrációjának biztosítása, az alapvető entitások létrehozása *.jdl fájl alapján, valamint az alapértelmezett felhasználókezelési funkciók előkészítése. Az által nyújtott generálás jelentősen hozzájárult az alkalmazás gyors előkészítéséhez és beüzemeléséhez.

3.2. Követelmény elemzés

3.2.1. Felhasználók

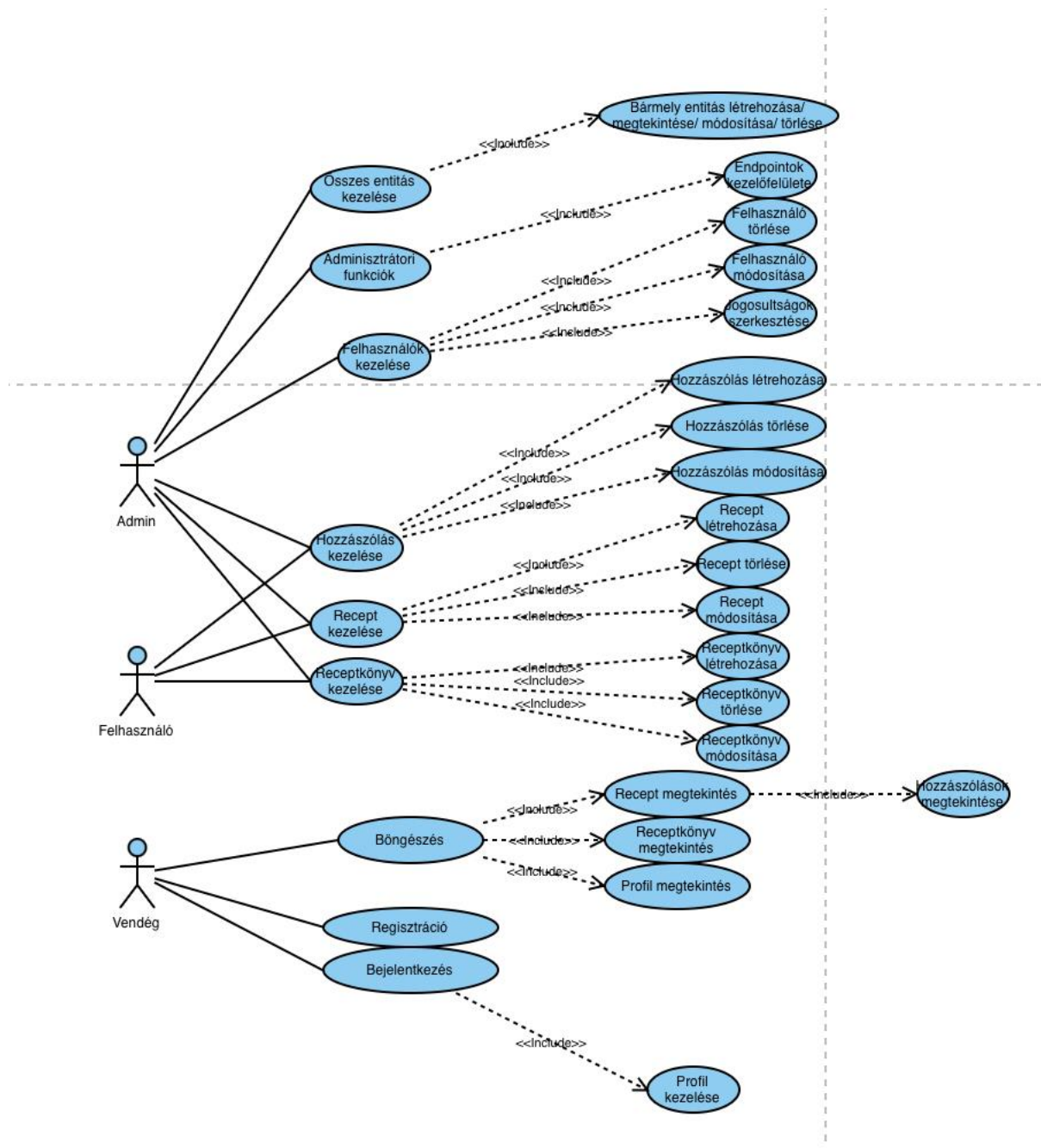
- Adminisztrátor
 - Összes entitáshoz hozzáférés, adminisztrátori, monitorizálás funkciók ellátására képes
 - Jogosultságok kezelése, felhasználó aktiválás
- Felhasználó
 - Extra jogosultsággal rendelkezik vendéghez képes. Új recept hozzáadása, hozzászólás írás, receptkönyv létrehozás és az ezekhez tartozó módosítás.
 - Saját adatainak módosítása
- Vendég
 - Csak receptek, receptkönyvek és profilok böngészésére képes, illetve bejelentkezésre és új felhasználó létrehozására képes.

3.2.2. Felhasználói követelmények

- Adminisztrátor képes:
 - Összes entitás létrehozására
 - Összes entitás módosítása magas szinten
 - Összes entitás adminisztrációs információit látni magas szinten
 - Összes entitás törlésére
 - Endpoint-ok látása, használata
 - Felhasználók létrehozása
 - Felhasználók adatainak módosítása
 - Felhasználók törlésére
 - Felhasználók adminisztrációs információit látni
 - Felhasználók jogosultságainak módosítása
 - Új recept létrehozása
 - Meglévő receptek módosítása
 - Új receptkönyv létrehozása
 - Meglévő receptkönyvek módosítása
- Felhasználó képes:
 - Új recept létrehozása

- Meglévő saját receptek módosítása
- Saját recept törlése
- Új receptkönyv létrehozása
- Meglévő receptkönyvek módosítása
- Saját receptkönyvek törlése
- Hozzászólás írása
- Saját hozzászólás módosítása
- Saját hozzászólás törlése
- Profil adatok módosítása
- Receptek megtekintése
- Receptkönyvek megtekintése
- Kereső használata
- Profilok megtekintése
- Vendég képes:
 - Receptek megtekintése
 - Receptkönyvek megtekintése
 - Kereső használata
 - Profilok megtekintése
 - Bejelentkezés/Regisztráció
 - Nyelvváltoztatás

3.2.3. Funkcionális követelmények use case diagrammal szemléltetve



ábra 34: Use Case diagram a funkcionális követelmények szemléltetésére

3.2.4. Nem funkcionális követelmények

Alkalmazás szempontjából legfontosabb szempont a kényelem. A felhasználó számára a kényelem biztosítása a legfontosabb, ami a könnyű kezelhetőség, szimplicitás és letisztultságban merül ki. Ne legyen szükség útmutatóra az applikáció használatához, minden egyértelmű legyen már az első alkalommal. Egyértelműen el vannak különítve az egymáshoz

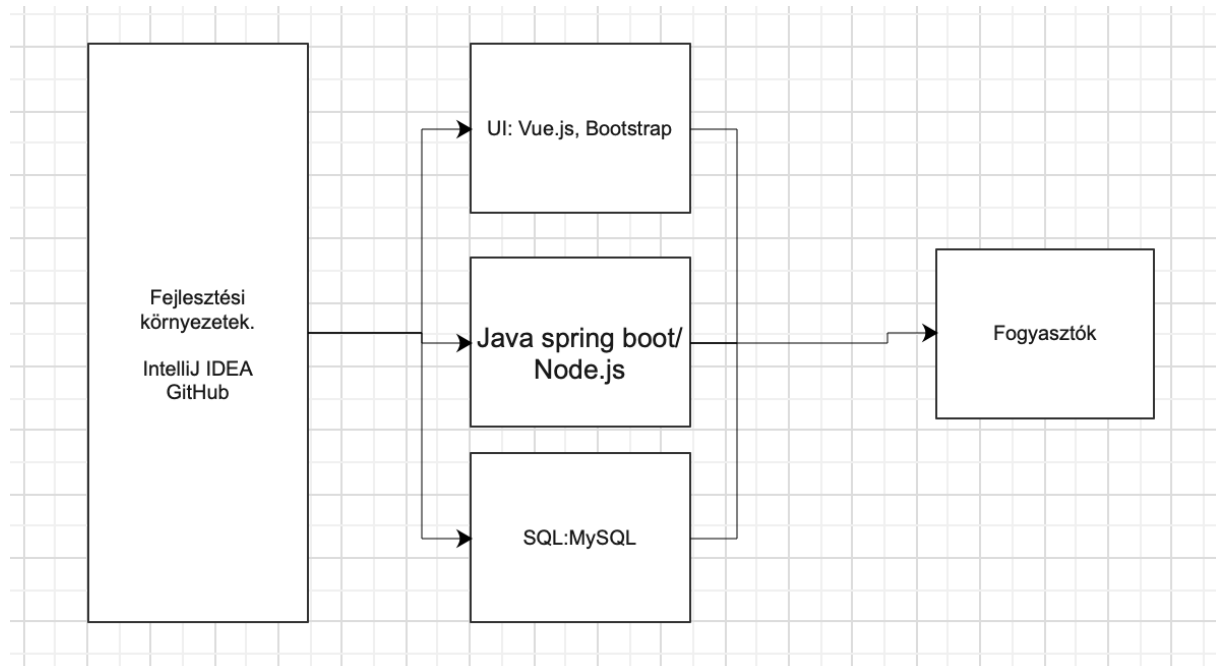
tartozó funkciók, nincsenek felesleges oldalak és lekérdezések, ezzel hozzájárulva a következő ponthoz.

Másodlagos szempont a hatékonyság, mivel egy recept alkalmazás, ezért feltételezhető, hogy magas számú étel, ételkönyv és profil lesz benne, ami nagyon magas overheaddel tud járni, ezért az adatmennyiség lekérdezések le vannak korlátozva, szerver oldali terhelés van.

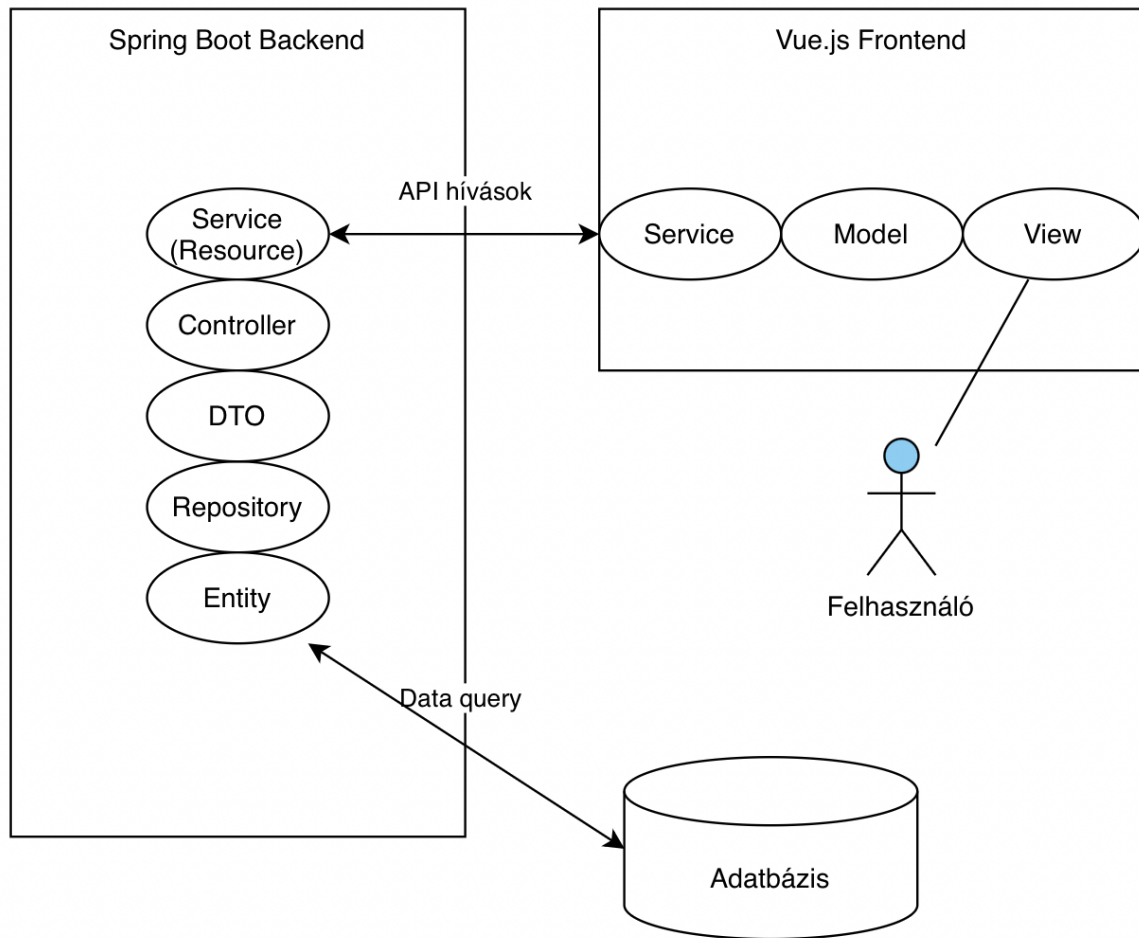
Továbbá még fontos megemlíteni a skálázhatóság kérdését. Az entitások, oldalak struktúrájának kiépítése során nagyon sok figyelem terelődött a továbbfejlesztésre. Nincsenek felesleges adattagok, a lehető legkevesebb adattárolással, a lehető legtöbb kihozására lett törekedve. Az entitások nagyon könnyen bővíthetőek, módosíthatóak a megfelelő struktúrának köszönhetően.

3.3. A szoftver logikai és fizikai szerkezetének leírása

3.3.1. Szerver és kliensoldali keretrendszer



3.3.2. Az alkalmazás felépítése

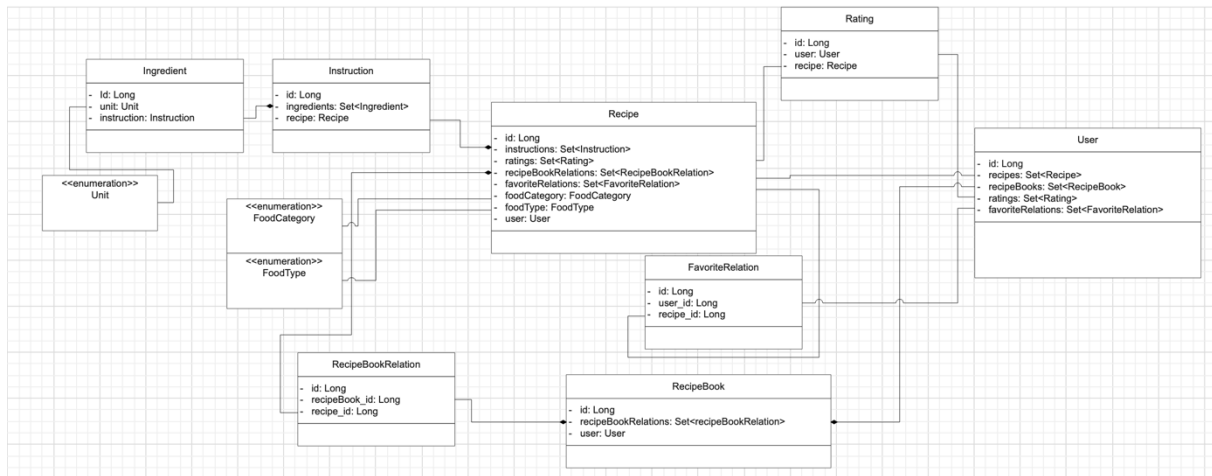


ábra 35: Alkalmazás felépítése diagrammal szemlélítve

Az adatbázisból data query-kkel tudjuk az adatokat lekérdezni, amivel már a saját entitás szerkezetünként lesz betöltve az adat. Ezekhez az entításokhoz tartozó lekérdezések repositorykban vannak definiálva. Következő lépés egy lekérdezés során a controller lesz, ahol az adat DTO-kká lesznek alakítva egy mapper segítségével. Utolsó lépésként a backend és frontend között servicek vannak. Frontenden szintén van egy megadott Model mindegyik entitáshoz, ami alapján lesz átvéve. Végezetül pedig az oldalakhoz tartozó View, amivel már közvetlenül kommunikál a felhasználó az alkalmazás használata során.

3.3.3. Adatbázis

3.3.3.1. Entitás UML diagram



ábra 36: Entitások és köztük lévő kapcsolat UML diagrammal szemlélítve

3.3.3.2. Entitás létrehozása

Adat entitások a domain fájlban vannak definiálva, ahol meg kell adni az összes hozzá tartozó tulajdonságot. Példa a Recept entitásról.

```
@Entity  @ Rupert Bence Gal *
@Table(name = "recipe")
@Cache(usage = CacheConcurrencyStrategy.READ_WRITE)
@SuppressWarnings("common-java:DuplicatedBlocks")
public class Recipe implements Serializable {
    @Id
    @GeneratedValue(strategy = GenerationType.SEQUENCE, generator = "sequenceGenerator")
    @SequenceGenerator(name = "sequenceGenerator")
    @Column(name = "id")
    private Long id;

    @Column(name = "title") 2 usages
    private String title;

    @Column(name = "description") 2 usages
    private String description;

    @Column(name = "ingredient_names") 2 usages
    private String ingredientNames;

    @Enumerated(EnumType.STRING) 2 usages
    @Column(name = "food_category")
    private FoodCategory foodCategory;
```

ábra 37: Entitás példa fontos annotációkkal ellátva

Annotációk:

- Entitás:
 - @Entity – annotáció jelzi, hogy az a java osztály JPA entitás és egy adatbázis táblát képvisel. Spring bootban szükséges, ha perzisztens entitásként akarjuk kezelni és lekérdezéseket akarunk lefolytatni rajta, módosítani ezzel JPA repositoryn keresztül.
 - @Table (name = 'tábla neve') – Megadja melyik adatbázis táblához lesz leképezve az entitás.
- Adattag:
 - @Column (name = 'oszlop neve') – Megadja az adatbázisban az oszlop nevét
 - @Enumerated – az enum típusok adatbázisban történő tárolására használjuk, megadható vele típus és beállítható hogyan legyen tárolva a mező
 - @ManyToOne – Az adott entitás egy másikkal való kapcsolatát írja le, attól függően melyik változatát adjuk meg az annotációnak. Ez lehet még ManyToMany, OneToOne, OneToMany.
 - @JsonIgnoreProperties – Jackson könyvtár része, serializáció szabályozását hivatott segíteni, megadható melyik tulajdonság kerüljön figyelmen kívül.
 - @Id – jelzi, hogy ez az entitás elsődleges kulcsa
 - @GeneratedValue - meghatározza a kulcs generálásának módját

3.3.3.3. Alkalmazásban definiált entitások

Az enum típusokat leszámítva mindegyik entitás rendelkezik egy szekvenciálisan kiosztott egyedi ID-vel, ami az elsődleges kulcs minden esetben és Long primitív típusként van megadva.

- Ingredient – Hozzávaló:
 - A name szöveg típus, amount az egész szám primitív típusok, unit az Unit típusú enum és az instruction pedig az Instruction-höz való kapcsolatot szimbolizálja, ami kompozícióval van megvalósítva.
- Instruction – Lépés:

- A title, description szöveg típus, requiredTime egész szám, ingredients egy hashset, ami a hozzá tartozó kompozícióval kapcsolódó hozzávalókat tárolja és a recipe pedig a Recipe típusú szintén ilyen kapcsolattal másik oldalról kapcsolódó recept.
- Rating – Hozzászólás:
 - Title, description, imageUrl szöveg típusok, a kép az elérési útvonalként van elmentve. Rate az értékelés értéke egész számként, createdAt egy ZonedDateTime java beépített osztály a kreálási időt hivatott jegyezni. Két kapcsolata van más osztályokhoz, egyik a recipe Recipe típus és másik a user User típus, amelyekre kompozícióval kapcsolódik.
- RecipeBook – Receptkönyv:
 - Title, theme, description, tags mind primitív szöveg típusok, createdAt ZonedDateTime beépített típus, láthatóságra egy isVisible boolean típussal. Továbbá kompozícióval kapcsolódik a Userre és ManyToMany kapcsolatot szimulálva egy segéd táblával egy HashSet-be mentve RecipeBookRelation típussal a recipeBookRelations mezőn aggregációs kapcsolat van Receptekkel, ami a benne lévő receptek listája.
- Recipe – Recept:
 - Title, description, ingredientsNames, imageUrl mind szöveg típusok, foodCategory és foodType saját enum típus FoodCategory és FoodType-val vannak megadva, createdAt ZonedDateTime. Maradék tulajdonság kapcsolatokat mutat más táblákkal. Kompozícióval kapcsolódik User felé, instructions, ratings, favoriteRelations viszont a recept felé van kompozícióban. Az utolsó kapcsolata pedig a recipeBookRelations, amely pedig aggregációssal jön létre.
- User – Felhasználó:
 - Login, password, firstName, lastName, langKey mind primitív szöveg típusok. Avatar egy egész szám, activated egy boolean, createdAt ZonedDateTime. Tartozik hozzá még egy authorities, ami egy szimpla asszociáció az Authority osztállyal.

Enum típusok:

- Unit – Egység:
 - Az Ingredient típushoz tartozik és szöveg típusként megadott, 3 lehetséges értéke lehet: PIECE, MILLILITRES, GRAM. Mint darab, milliliter és gram.
- FoodCategory – étel kategória:
 - A Recipe típusnál van használva és az ételek konyháját jelenti (cuisine). Sok értéke lehet ezek közül néhány: HUNGARIAN, JAPANESE, MEXICAN. Mint magyar, japán és mexikói konyha.
- FoodType – étel típus:
 - A Recipe típusnál van használva és az ételek tálalására, felszolgálására vagy a fogyasztására utalnak. Szintén több értéke lehet például: APPETIZER, SALAD, STEW. Mint előétel, saláta vagy pörkölt.

3.3.3.4. DTOk (Data Transfer Object)

A frontendre továbbított adatok védelme és korlátozása érdekében DTO-k kerültek bevezetésre. Ezek az objektumok az adatbázisból érkező entitások lekérdezése után, a Service rétegben kerülnek átmappelésre az entitásokról a DTO-kra. Az így létrejövő Objektumok továbbításra kerülnek a kliensoldalra, biztosítva ezzel az adatok megfelelő formátumát és szűrését.

A legtöbb esetben a DTO-k szerkezete azonos az entitásaikkal, azonban speciális esetekben a DTO-k kiegészülhetnek extra attribútumokkal, amelyek nem részei az eredeti entitásnak. Ezek az extra tulajdonságok célzott funkcionalitást biztosítanak a frontend számára. Az alábbiakban kizárólag ezekre az esetekre fogok kitérni.

- Recept:
 - isFavorite: egy logikai adat, az aktuálisan bejelentkezett felhasználóra vonatkozó információ és kedvelést reflektálja
 - favoriteCount: Long típusú, a kedvelések szummáját mutatja
 - requiredTimeSum: Long típusú és a recepthez tartozó instrukciók elkészítési idejének összege
 - totalRating: Double típus, a felhasználók által pontozások átlaga
- Receptkönyv:

- isRecipeInList: logikai változó, ami az adott receptre adja meg, hogy a könyvben szerepel-e
- recipeImages: szöveges típusból álló lista és a könyv előnézetéhez tartalmaz képeket a tartalmazott receptekből
- AdminUserDTO:
 - Az alapértelmezett DTO a felhasználók számára.
- Profil:
 - A felhasználók egy tovább gondolt változata.
 - recipes: Recept típusú lista, a felhasználó által hozzáadott recepteket hivatkozza
 - recipeBooks: Receptkönyv típusú lista, a felhasználó által hozzáadott receptkönyveket hivatkozza
 - sumRecipes: Long típusú és a recipes lista méretét adja meg
 - sumRecipeBooks: Long típusú és a recipeBooks lista méretét adja meg (szükséges az két szumma extra mezőre a pagination miatt a profil oldalon)

3.3.3.5. Adatbázis végpontok létrehozása

Minden entitáshoz tartozik három fájl, hogy végpontot tudjunk létrehozni, ezek a Resource, Service és Repository. Az alábbi példák a legutóbb létrehozott receptek listájának lekérdezését fogják szemléltetni lépésről-lépésre.

Legfelső szinten helyezkedik el a Resource fájl, ez hivatott a kliensoldalról érkező kéréseket fogadni, illetve az adatbázisból érkező, már leszűrt, formázott legtöbb esetben DTO-kat (ez nem zárja ki a primitív típus vagy entitás válaszokat sem) továbbítani szintén a kliensoldal felé. Itt tudunk bármilyen végpontot, azaz a mappinget definiálni függően attól, hogy milyen célra szánjuk és ez fogja megadni majd az elérési útvonalat.

```
@GetMapping("queries/latest")
public List<RecipeDTO> getLatestRecipes() {
    LOG.debug("REST request to get recipes with the latest date");
    // Fetch all the latest recipes
    return recipeService.findLatestRecipes();
}
```

ábra 38: Resource fájl példa

Középső szinten helyezkedik el a Service, ahol a már adatbázisból lekérdezéssel kinyert bármilyen adatot átalakítjuk, formázzuk, szűrjük vagy ha a frontendről érkezik adat akkor az adatbázisban eltárolható formába alakítjuk. Bizonyos esetekben az adatbázisra leképezhető lekérdezések itt kerülnek megépítésre, mint például a receptek szűrésénél. Ebben a rétegben történik meg az adatok védelmére szolgáló DTO-kba való konvertálás, mappelés vagy az adatbázisba eltárolható entitások összeállítása, tényleges kapcsolatokkal.

```
@Transactional(readOnly = true) 1 usage 1 Rupert Bence Gal
public List<RecipeDTO> findLatestRecipes() {
    LOG.debug("Request to get most lately added Recipes");
    return recipeRepository.findLatestRecipes().stream().map(recipeMapper::toDto).collect(Collectors.toCollection(LinkedList::new));
}
```

ábra 39: Service fájl példa

Legalsó szinten a Repository, azaz az adatbázis kezelés réteg áll. Ezen a szinten történnek az adatbázissal kapcsolatos közvetlen műveletek, mint például adatok lekérdezése, mentése. Az itt létrehozott lekérdezések biztosítják az adatokat a felsőbb rétegek számára.

```
@Query("select r from Recipe r order by r.createdDate desc")
List<Recipe> findLatestRecipes();
```

ábra 40: Repository fájl, lekérdezés példa

3.3.3.6. Resource-ok által kiszolgált endpointok

3.3.3.6.1 Minden entitásra megegyező végpontok:

- /api/{entity}s/{id}
- /api/{entity}s

3.3.3.6.2 További az alapl műveletektől, eltérő, speciális végpontok

Recipe:

- /api/recipes/{id}/ratings
- /api/recipes/search
- /api/recipes/queries/same_food_category/{food_category}
- /api/recipes/queries/latest
- /api/recipes/queries/highest_favorite_count

RecipeBook:

- /api/recipe-books/{recipeBookId}/{recipeId}
- /api/recipe-books/status/{recipeBookId}
- /api/recipe-books/my-recipe-books
- /api/recipe-books/find-in-recipe/{recipeId}

RecipeBookRelation:

- /api/recipe-book-relations/all-by-recipe-book-id/{recipeBookId}
- /api/recipe-book-relations/delete/{recipeId}/{recipeBookId}

Rating:

- /api/ratings/{recipeId}

Ingredient:

- /api/ingredients/unique_ingredients

Image:

- /api/images/upload
- /api/images/{upload_date}/{image_name}

Account:

- /api/register
- /api/account/change-password
- /api/authenticate

3.3.3.7. Fontosabb, komplexebb metódusok

Recipe:

- searchRecipes ('/recipes/search'): A recept keresés metódusa, paraméterei tartalmazzák a keresési feltételeket: cím és három lista mind szöveges típussal, a listák azok az étel kategória, -típus és alapanyagot jelentik. Az utolsó típus pedig Pageable, ami tartalmazza a rendezés mezőjét és irányát. A lekérdezés

query builder segítségével történik, amin belül a megadott paraméterek alapján folyik le majd a tényleges lekérdezés. Minél több paraméterünk van, annál nagyobb lesz a válasz halmaz, tehát a szűrés ÉS kapcsolatokkal történik.

Image:

- `uploadImage ('/images/upload')`: Képfeltöltésnél alkalmazott metódus, paramétere egy `MultipartFile` típusú fájl. A fájlok elmentése egy erre külön a gyökér könyvtárban létrehozott mappában, az `images`-be történik. A képek kategorizálása dátumok alapján történik, formátuma `'YYYY-MM-DD'`. Minden kép neve megváltoztatásra kerül és egy véletlenszerűen generált UUID-t kap helyette.
- `getPhoto ('images/upload_date}/{image_name}')`: Az adatbázisban szereplő, feltöltött képek ezen a végponton érhetőek el, viszont szükség van a kép feltöltésének dátumára, illetve a kép feltöltéskor megkapott nevére, melyeket paraméterként kell az url-be illeszteni.

Ingredient:

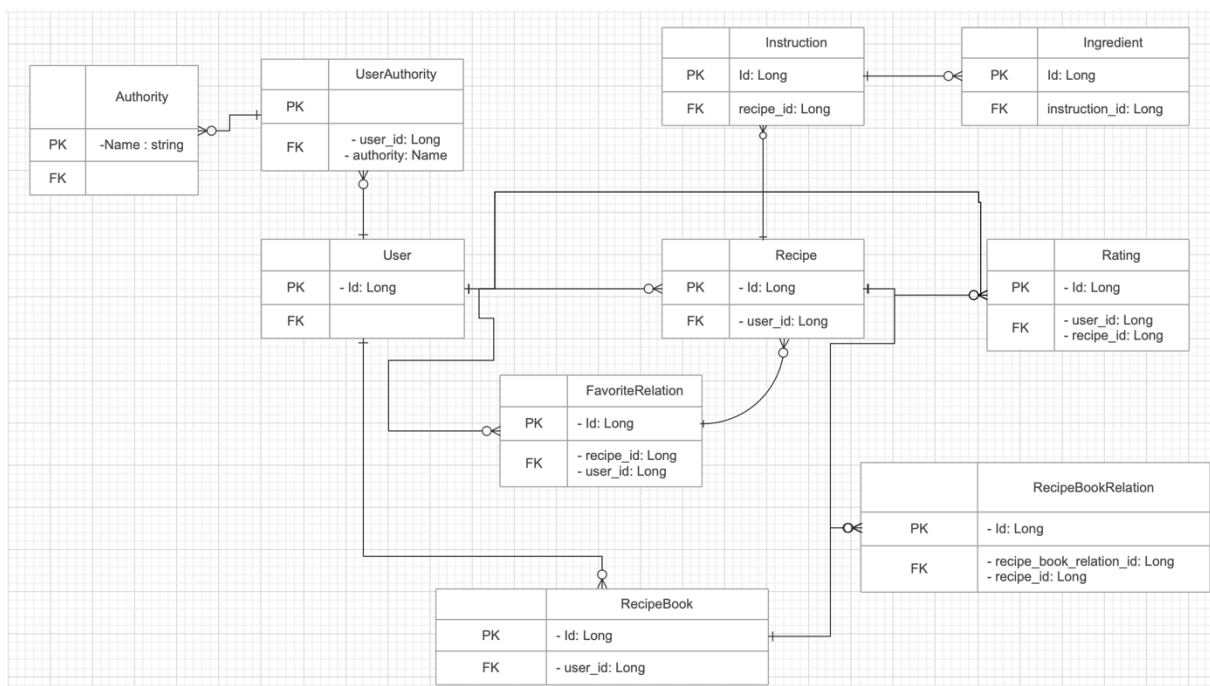
- `searchAllRecipeBooks(/recipe-books)`: Ez a kereső és egyben összes recepteskönyvet lekérdező metódus, nem kötelező paraméternek pedig át lehet adni a címet, témát, tageket vagy a leírást. A megadott paraméterek segítségével viszont már kereső metódus fog végbemenni.

3.3.3.8. Az alkalmazásban használt adattáblák

Minden táblában minden entitásnak van egy egyedi saját elsődleges kulcsa, amik Long-ként kerülnek eltárolásra és az adattáblák közötti kapcsolatot ezek hivatottak biztosítani. A User entitásnál megtalálható az adminisztrációt segítő 4 fő adattag: `createdBy(String)`, `createdAt(Instant)`, `lastModifiedBy(String)`, `lastModifiedDate(Instant)`. A fennálló többi Objektum esetében a Recipe, RecipeBook, Rating-t leszámítva nem találhatók meg ezek, hiszen az alkalmazás szempontjából nem képez extra értéket és optimalizációs szempontból jelentős, a felsoroltaknál viszont a `createdAt` tulajdonság van csak eltárolva `ZonedDateTime` típussal.

Az adattáblákban minden entitás a 3.3.2.2 pontban leírtak szerint van elmentve leszámítva a táblák közötti kapcsolatot (hivatkozások más entitásra), azok automatikusan össze vannak kapcsolva a ID-k alapján. A ManyToOne kapcsolatoknál a Many oldalon nincsen eltárolva az adott táblában, másik oldalon a Many-hez való kapcsolódás elsődleges kulccsal van. OneToOne kapcsolat csak Rekord és User között van, ezek a rekord adattáblájában vannak tárolva user_id alatt. ManyToMany kapcsolatok esetén segéd táblák lettek definiálva, ahol mind a két táblát összekapcsoló kulcsok vannak eltárolva és ManyToOne kapcsolatokra van mindkét fő táblához kapcsolva.

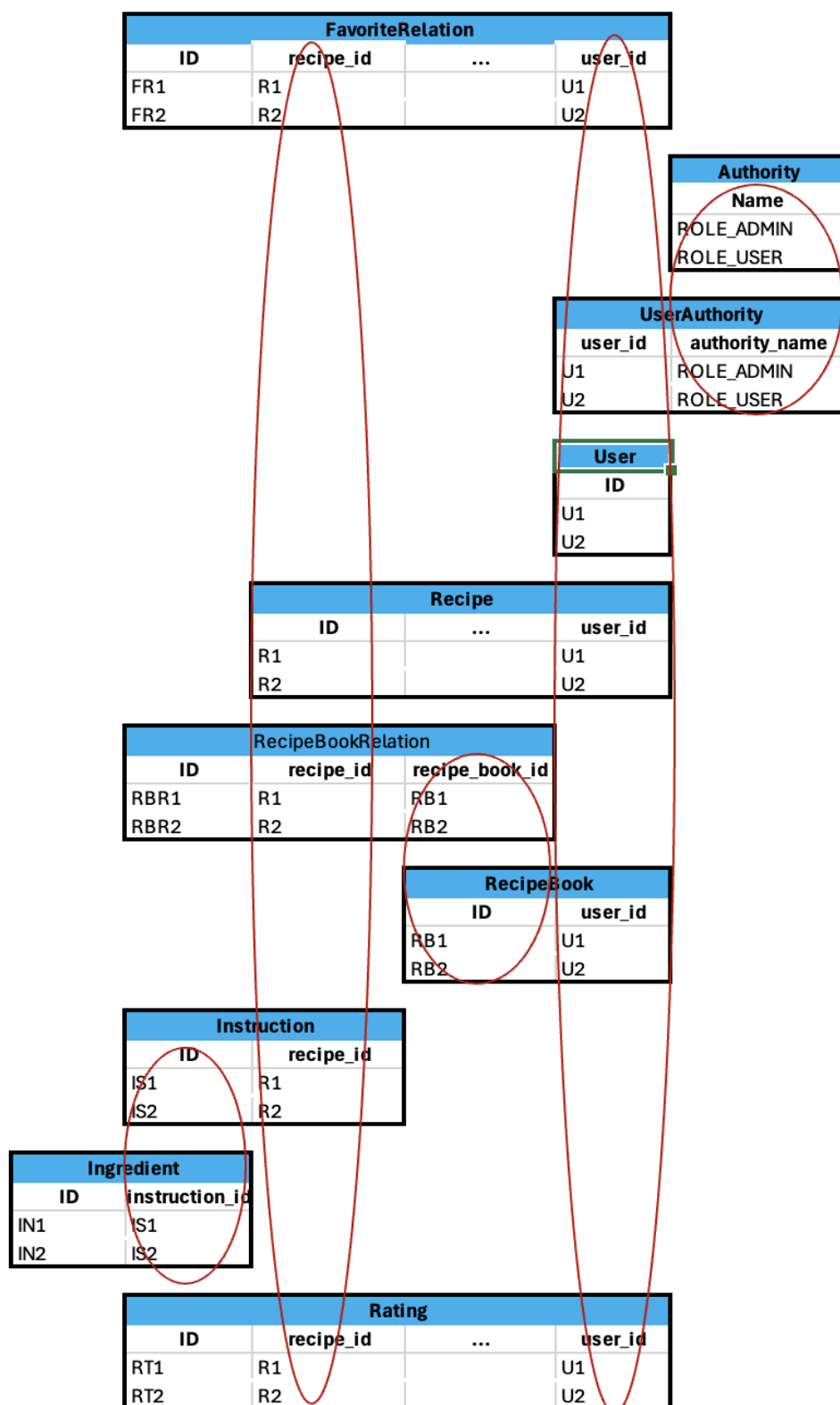
3.3.3.9. Kulcs Kapcsolatok



ábra 41: Kulcs kapcsolatok

User és Authority táblák között 1..1-es kapcsolat van, az összes többi tábla 0*..1 kapcsolatot alkalmaz és az minden esetben az elsődleges kulcsokra hivatkozva.

3.3.3.10. Adatkapcsolati modell



ábra 42: Adatkapcsolati modell

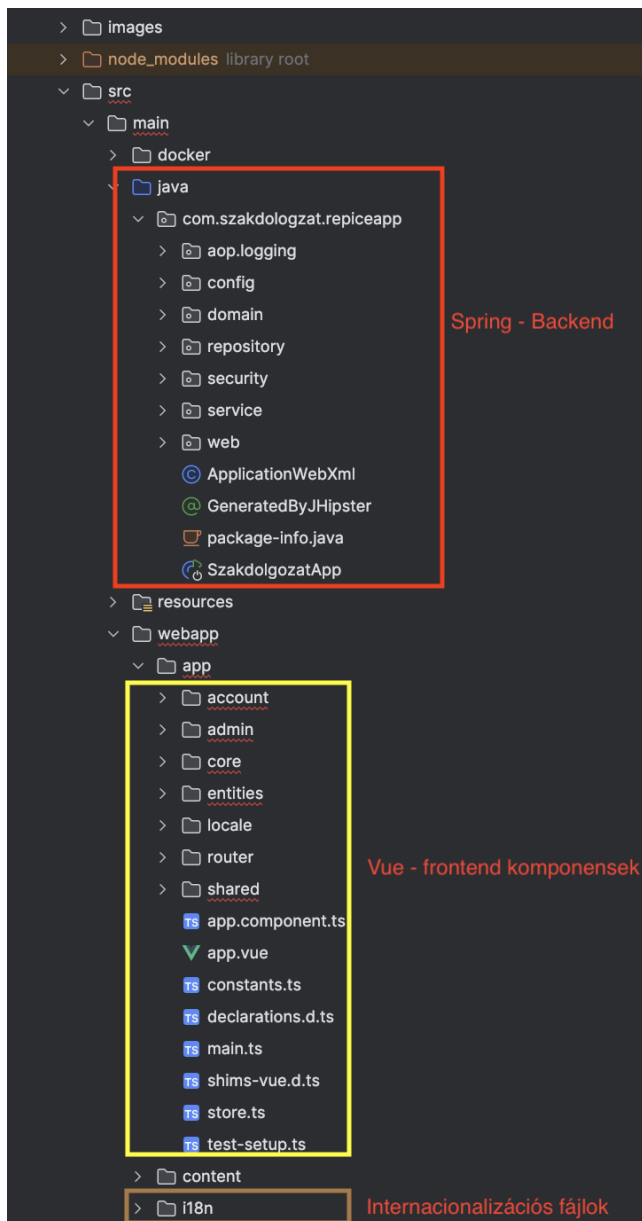
Minden táblán az Authority-t leszámítva az elsődleges kulcsok Long-ként vannak létrehozva, és szekvenciálisan van generálva. Authority esetében szöveges típus. Ezekkel az elsődleges kulcsokkal hivatkoznak egymásra a nézetek.

3.3.3.11. *.csv fájlok felépítése

```
id;rate;title;description;image_url;recipe_id;user_id;created_date
1;2;Needs Improvement;The flavor was too strong and overwhelming.;http://localhost:8080/images/database/sweet-and-sour-chicken-2.jpg;1;4;2024-11-06T00:00:00
2;3;Decent;Fairly memorable, but not extraordinary.;http://localhost:8080/images/database/sweet-and-sour-chicken-3.jpg;1;9;2024-11-06T00:00:00
3;1;Disappointing;Disconnected flavors, not very enjoyable.;http://localhost:8080/images/database/spaghetti-carbonara-2.jpg;2;6;2024-11-05T00:00:00
```

ábra 43: Rating adatbázis adatai (részlet)

3.3.4. Projekt struktúrája



A főkönyvtárban elhelyezkedő src fájl tartalmazza egyaránt a frontend és a backend fájljait, illetve a tesztek is. A spring fájlrendszerben konvenció szerint vannak rétegek szerint szedve az összetartozó osztályok. Négy lényeges könyvtár van, domain – az entitásoknak, repository – a közvetlen adatbázis lekvásoknak, service és web – DTOk és Controllereknek. A backendhez tartozó tesztek a src/test – ban találhatóak.

A kliensoldalon található webapp/core tartalmazza az oldalak megjelenítéséhez tartozó vue fájlokat és komponens fájljaikat. l18n tartalmazza a fordításhoz szükséges fájlokat, nyelvekre bontva. Minden tesztfájl az tesztelni kívánt könyvtárban helyezkednek el.

A képeknek gyökérkönyvtárban van egy külön images mappa, azon belül a database-ben található fájlok a receptekhez tartozó képek.

3.4. Tesztelés

3.4.1. Tesztelési terv és tesztelési eredmények

A projekt tesztelése teljeskörűen lefedi az alkalmazás működését. Mivel az alkalmazás több rétegből épül fel, minden egyes réteg alapos tesztelésen esett át. Az alábbi táblázatokban elsősorban a legfontosabb és leggyakrabban használt serveroldali integrációs tesztek szerepelnek. Minden entitás és azok metódusai, valamint a lényegesebb Service is egységtesztelésen mentek keresztül.

A kliensoldali tesztelés során kiemelt figyelmet fordítottam az összes fájlra és funkciójának alapos működésbeli vizsgálatára.

3.4.1.1. Létrehozás

táblázat 1: Létrehozás tesztek

Tesztesetek száma	Teszt leírása	Bemenetek	Várt eredmény	Eredmény
1-1	Új alapanyag létrehozása.	Név: "name" Mennyiség: 1 Egység: GRAM	Új alapanyag létrejön.	Létrejött az alapanyag,

1-2	Létező ID-vel új alapanyag létrehozása.	ID: 1 Név: "name" Mennyiség: 1 Egység: GRAM	Nem jön létre új elem mivel már létezik az ID.	Nem jött létre új elem és hiba választ kaptunk.
2-1	Új instrukció létrehozása.	Cím: „title” Leírás: „description” Idő: 1	Új instrukció létrejön.	Létrejött az instrukció.
2-2	Létező ID-vel új instrukció létrehozása	ID:1 Cím: „title” Leírás: „description” Idő: 1	Nem jön létre új instrukció mivel már létezik az ID.	Nem jött létre új elem és hiba választ kaptunk.
3-1	Új recept létrehozása.	Cím: „title” Leírás: „description” Típus: APPETIZER Kategória:CHINESE	Új recept létrejön.	Létrejött az recept.
3-2	Létező ID-vel új recept létrehozása	ID:1 Cím: „title” Leírás: „description” Típus: APPETIZER Kategória:CHINESE	Nem jön létre új recept mivel már létezik az ID.	Nem jött létre új elem és hiba választ kaptunk.
4-1	Hozzászólás létrehozása recepthez	Érték:1 Leírás:”description”	Létrejön a hozzászólás a megfelelő	Létrejött a hozzászólás és a kapcsolat

		Recept ID:1	recepthez kötve.	a recept között.
4-2	Hozzászólás létrehozása rossz recepthez.	Érték:1 Leírás:" description" Recept ID:0	Nem jön létre a hozzászólás és hibaüzenetet kapunk.	Hibaüzenet és nem jött létre a hozzászólás.
5-1	Új receptkönyv létrehozása	Cím: „title” Téma: „theme” Leírás:" description"	Létrejön a receptkönyv a megfelelő adatokkal.	Létrejött az új receptkönyv, elmentve az adatbázisban.
5-2	Létező ID-vel új receptkönyv létrehozása	ID:1 Cím: „title” Téma: „theme” Leírás:" description"	Nem jön létre új receptkönyv mivel már létezik az ID.	Nem jött létre új elem és hiba választ kaptunk.
6-1	Új felhasználó létrehozása	Felhasználónév: „johndoe” Vezetéknév: „doe” Keresznév: „john”	Létrejön a felhasználó a megfelelő adatokkal.	Létrejött az új felhasználó, elmentve az adatbázisban.
6-2	Létező felhasználónévvel új felhasználó létrehozása	Felhasználónév: „johndoe” Vezetéknév: „doe” Keresznév: „john”	Nem jön létre új felhasználó mivel már létezik a felhasználónév.	Nem jött létre új elem és hiba választ kaptunk.
7-1	Kép feltöltés.	Fájl: Képfájl	Feltöltődik a kép és elmentődik a	Létrejönnek a tároló fájlok

			megfelelő fájlrendszerben.	és a kép el lett mentve.
--	--	--	-------------------------------	-----------------------------

3.4.1.2. Módosítás

táblázat 2: Módosítás tesztek

Tesztesetek száma	Teszt leírása	Bemenetek	Várt eredmény	Eredmény
1-1	Létező alapanyag módosítása.	ID:1 Név: "updated_name" Mennyiség: 2 Egység: MILLILITRES	Már meglévő alapanyag adatai módosulnak.	Módosultak az alapanyag adatai.
1-2	Nem létező alapanyag módosítása.	ID: 0 Név: "updated_name" Mennyiség: 2 Egység: MILLILITRES	Nem módosul semmi, hibaüzenetet kapunk.	Nem módosult az alapanyag és hibaüzenetet kaptunk.
1-3	Létező alapanyag részleges módosítása.	ID:1 Név: "updated_name" Mennyiség: 1 Egység: MILLILITRES	Már meglévő alapanyag adatai részlegesen módosulnak.	Módosultak az alapanyag adatai.
2-1	Létező instrukció módosítása.	ID:1 Cím: „updated_title” Leírás:	Már meglévő instrukció adatai módosulnak.	Módosultak az instrukció adatai.

		„updated_description” Idő: 2		
2-2	Nem létező instrukció módosítása.	ID: 0 Cím: „updated_title” Leírás: „description” Idő: 2	Nem módosul semmi, hibaüzenetet kapunk.	Nem módosult az instrukció és hibaüzenetet kaptunk.
2-3	Létező instrukció részleges módosítása.	ID:1 Cím: „updated_title” Leírás: „updated_description” Idő: 2	Már meglévő instrukció adatai részlegesen módosulnak.	Módosultak az instrukció adatai.
3-1	Létező recept módosítása.	ID:1 Cím: „updated_title” Leírás: „updated_description” Típus: DESSERT Kategória:ITALIAN	Már meglévő recept adatai módosulnak.	Módosultak a recept adatai.
3-2	Nem létező recept módosítása.	ID:0 Cím: „updated_title” Leírás: „updated_description” Típus: DESSERT Kategória:ITALIAN	Nem módosul semmi, hibaüzenetet kapunk.	Nem módosult a recept és hibaüzenetet kaptunk.
3-3	Létező recept részleges módosítása.	ID:1 Cím: „updated_title” Leírás: „description”	Már meglévő recept adatai részlegesen módosulnak.	Módosultak a recept adatai.

		Típus: DESSERT Kategória: ITALIAN		
4-1	Létező hozzászólás módosítása.	Érték:2 Leírás: "updated_description" Recept ID:1	Már meglévő hozzászólás adatai módosulnak.	Módosultak a hozzászólás adatai.
4-2	Nem létező hozzászólás módosítása.	Érték:2 Leírás:" description" Recept ID:1	Nem módosul semmi, hibaüzenetet kapunk.	Nem módosult az hozzászólás és hibaüzenetet kaptunk.
4-3	Létező hozzászólás részleges módosítása.	Érték:2 Leírás: "updated_description" Recept ID:1	Már meglévő hozzászólás adatai részlegesen módosulnak.	Módosultak a hozzászólás adatai.
5-1	Létező receptkönyv módosítása.	ID:1 Cím: „updated_title” Téma: „updated_theme” Leírás:” updated_description”	Már meglévő receptkönyv adatai módosulnak.	Módosultak a receptkönyv adatai.
5-2	Nem létező receptkönyv módosítása.	ID:0 Cím: „updated_title” Téma: „updated_theme” Leírás:” updated_description”	Nem módosul semmi, hibaüzenetet kapunk.	Nem módosult az receptkönyv és hibaüzenetet kaptunk.

5-3	Létező receptkönyv részleges módosítása.	ID:1 Cím: „title” Téma: „updated_theme” Leírás: ” updated_description”	Már meglévő receptkönyv adatai részlegesen módosulnak.	Módosultak a receptkönyv adatai.
6-1	Létező felhasználó részleges módosítása.	ID:1 Felhasználónév: „jamesdon” Vezetéknév: „johndoe” Keresznév: „john”	Létrejön a felhasználó a megfelelő adatokkal.	Létrejött az új felhasználó, elmentve az adatbázisban.
6-2	Nem létező felhasználó módosítása.	ID:0 Felhasználónév: „johndoe” Vezetéknév: „don” Keresznév: „james”	Nem módosul semmi, hibaüzenetet kapunk.	Nem módosult az felhasználó és hibaüzenetet kaptunk.
6-3	Létező felhasználó részleges módosítása.	ID:1 Felhasználónév: „jamesdon” Vezetéknév: „don” Keresznév: „john”	Már meglévő felhasználó adatai részlegesen módosulnak.	Módosultak a felhasználó adatai.

3.4.1.3. Törlés

táblázat 3: Törlési tesztek

Tesztesetek száma	Teszt leírása	Bemenetek	Várt eredmény	Eredmény
-------------------	---------------	-----------	---------------	----------

1-1	Létező alapanyag kitörlése.	ID:1	Meglévő alapanyag kitörlése.	Létező alapanyag kitörlődött.
2-1	Létező instrukció kitörlése.	ID:1	Meglévő instrukció kitörlése.	Létező instrukció kitörlődött.
3-1	Létező recept kitörlése.	ID:1	Meglévő recept kitörlése.	Létező recept kitörlődött.
4-1	Létező hozzászólás kitörlése.	ID:1	Meglévő hozzászólás kitörlése.	Létező hozzászólás kitörlődött.
5-1	Létező receptkönyv kitörlése.	ID:1	Meglévő receptkönyv kitörlése.	Létező receptkönyv kitörlődött.
6-1	Létező felhasználó kitörlése.	Felhasználónév: „johndoe”	Meglévő felhasználó kitörlése.	Létező felhasználó kitörlődött.

3.4.1.4. Lekérdezések

táblázat 4: Lekérdezés tesztek

Tesztesetek száma	Teszt leírása	Bemenetek	Várt eredmény	Eredmény
1-1	Recept keresés a megadott	ID:1 Cím: „title” Leírás: „description”	Visszakapott válasz listában paraméternek	Visszakapott recept kielégíti a követelményeket.

	paraméterek alapján	Típus: APPETIZER Kategória:CHINESE	megfelelő recept.	
1-2	Ugyanolyan kategóriás receptek lekérdezése.	Kategória: CHINESE	Csak a megadott kategóriában lévő receptek jönnek válaszban.	Csak a paraméterben megadott kategóriába tartozó receptek jöttek válaszban.
1-3	Legrégebbi receptek lekérdezése.	Nincs.	Legrégebbi feltöltési dátummal rendelkező receptek növekvő sorban.	Visszaadott lista helyesen van rendezve és a legrégibbi receptek.
2-1	Egyedi alapanyagok listájának lekérdezése.	Nincs. (Adatbázisban két megegyező névvel rendelkező alapanyag van)	Két különböző elem van a válaszlistában.	Csak különböző alapanyag van a listában és csak két elem van.

3.4.1.5. Használt eszközök

JUnit 5 és MapStruct segítségével jelentős mennyiségű egységtesztet sikerült generálnom a Java nyelvhez, melyek elsősorban az alapl műveletekre koncentráltak. A meglévő tesztekhez jelentős kiegészítésekre volt szükség a további implementált metódusok teszteléséhez. Az integrációs teszteket a Spring Test Context keretrendszer segítségével hajtottam végre. A JHipster generátorral az alapl műveletek automatikusan generálódtak, de itt is, akárcsak JUnit 5 esetében, szükség volt jelentős pótlásokra a teljes funkcionalitás teszteléséhez. A kliensoldali

tesztelést Jest keretrendszer alkalmazásával végeztem el, míg a működésbeli teszteket Sinon használatával valósítottam meg.

3.4.1.6. Tesztek futtatása

Az összes teszt futtatásához egyaránt a gyökérkönyvtárban szükséges parancssori parancsokat lefuttatni, ezen felül semmi mást nem kell csinálni. A szerveroldali tesztek az alábbi paranccsal futtathatók:

mvn verify

A kliensoldali tesztek pedig a következővel:

npm test

Mind a két tesztelés végén látható lesz a végeredmény a tesztesetekre vonatkozóan, részletesen lefedve. Abban az esetben, ha minden rendben ment akkor láthatjuk a kódlefedettséget a kliensoldalra vonatkozóan.

Szerveroldali tesztek futtatása után szükség lehet a generált class osztályok körlésére és egy friss generálásra, mivel összeakadhatnak a fájlok és futás idejű hibát kapunk, míg le nem futtatjuk, utána minden visszaáll az eredeti állásra:

mvn clean

4. Összefoglalás és további fejlesztési lehetőségek

4.1. Összefoglalás

A dolgozat célja egy olyan innovatív alkalmazás megtervezése és megvalósítása volt, amely áthidalja a mindennapi főzés során felmerülő kihívásokat és megkönnyíti a felhasználók számára az ételtervezést, receptkeresést és -elkészítést. Az alkalmazás egyszerű és felhasználóbarát felülete lehetőséget biztosít arra, hogy az emberek gyorsan és hatékonyan találjanak étkezési alkalmakhoz, szokásaikhoz, ízlésükhöz, diétájukhoz, valamint számukra elérhető alapanyagokhoz igazodó recepteket. Emellett az applikáció lehetővé teszi a receptgyűjtemények közötti böngészést is vagy éppen azok létrehozását, személyre szabását.

A fejlesztés során alkalmazott technológiai megoldások és funkciók, például az intuitív keresőmotor, a receptszűrési lehetőségek, a receptmegosztási funkció, mind azt a célt szolgálják, hogy a felhasználók időt és energiát takarítsanak meg, és a főzés folyamata minél gördülékenyebb legyen. A dolgozat részletesen bemutatta ezeket a funkciókat, valamint a felhasználói élmény optimalizálására alkalmazott módszereket és megfontolásokat.

Összességében a dolgozat rávilágít arra, hogy egy jól felépített, modern applikáció hatékony támogatást nyújthat a napi étkezés előkészítésében és az egészséges táplálkozás fenntartásában. Az elkészült alkalmazás nemcsak a gyors receptkeresésben és az egyéni igények szerinti szűrésben segít, hanem a főzéshez szükséges folyamatokat is leegyszerűsíti, ezáltal hozzájárulva a felhasználók kényelmesebb és egészségesebb életmódjához.

4.2. Továbbfejlesztési lehetőségek

Az applikáció számos továbbfejlesztési lehetőséggel rendelkezik, amelyekkel tovább lehet növelni felhasználói élményt akár legyen szó kényelmi funkciókról, akár az ételekre vonatkozó átfogóbb információkról, vagy akár működésbeli változásokról. Alább kategóriákba fogok szedni pár ötletet, amit tovább lehet gondolni, pontosítani.

- Felhasználó kezelés: Felhasználó aktiváció és email cím funkció befejezése, jelenleg csak létező mezők, de nincsenek tényleges funkcionalitások mögöttük. Felhasználókra vonatkozó extrabeállítások, itt lehet gondolni például megjegyzett mérési egységekre.
- Receptekre vonatkozó információk: Rengetek plusz információval el lehet látni recepteket, mivel jelenleg csak az esszenciális információk szerepelnek. Ilyen lehet például a tápérték számítás, allergének, tálalás mennyiségére szétbontható alapanyag mennyiségek, instrukciókra vagy egész recept elkészítésére vonatkozó szükséges eszközök.
- Jelenlegi funkciók fejlesztése: A kereső tovább gondolása, jelenleg ugyanis minden megadott feltétel csak bővíteni fogja az eredmény listát, ezt viszont szét lehetne választani, hogy keresni lehessen arra, amit szeretnénk a receptekben és el lehessen dönteni azt, amit szeretnénk elkerülni.
- Új funkciók: Bevásárló lista készítés nagy újítás lehetne és tökéletesen beleillene az jelenlegi alkalmazásba. Továbbá értéket képezhet még a recept megosztása vagy épp exportálása, nyomtatásra előkészített formátum készítése.

5. Irodalomjegyzék

- [1] „Java SE 8 Archive Downloads”

Link: <https://www.oracle.com/java/technologies/javase/javase8-archive-downloads.html> [Elérés dátuma: 2024.12.01.]

- [2] “JUnit 5”

Link: <https://junit.org/junit5/>

- [3] “Jest”

Link: <https://jestjs.io>

- [4] “Node.js”

Link: <https://nodejs.org/en>

- [5] „Maven – Introduction (apache.org)”

Link: <https://maven.apache.org/what-is-maven.html>

- [6] „Java”

Link: <https://www.java.com/en/download/manual.jsp>

- [7] „IntelliJ IDEA”

Link: <https://www.jetbrains.com/idea/download/?section=mac>

- [8] „Vue.js”

Link: <https://vuejs.org>

- [9] „Spring Boot”

Link: <https://spring.io>

- [10] “JHipster”

Link: <https://www.jhipster.tech>