**Architektúra-terv**

**1. Architektúra**

**1.1 Áttekintés**

A rendszer három fő komponensből áll:

* **Frontend**: AngularJS-alapú kliensoldali alkalmazás, amely felelős a felhasználói interakciók kezeléséért.
* **Backend**: C# nyelven, Entity Framework használatával fejlesztett REST API, amely az üzleti logikát és az adatkezelést biztosítja.
* **Adatbázis**: Relációs adatbáziskezelő rendszer: MySQL, amely az adatok tárolását és lekérdezését végzi.

Az alkalmazás Azure-felhőalapú infrastruktúrára kerül telepítésre, amely biztosítja az ingyenes és skálázható futtatási környezetet.

**1.2 Fő komponensek működése**

* **Frontend**: HTTP protokollon keresztül kommunikál a backenddel REST API-k hívásával. Emellett a felhasználói interface megjelenítésért felel.
* **Backend**: Az API hívások alapján adatokat kér le/módosít az adatbázisban, majd JSON formátumban válaszol a frontendnek.
* **Adatbázis**: A MySQL adatbázisban tárolódnak a felhasználói adatok, alkalmazásadatok, és az egyéb konfigurációk.

**2. Diagramtervek**

**2.1 Komponensdiagram**

Az alábbi ábra bemutatja az architektúra komponenseit és azok közötti kapcsolatrendszert:

A diagram of a computer program

Description automatically generated

**2.2 Telepítésiszolgáltatások**

1. **Frontend**: Azure Static Web Apps szolgáltatás segítségével kerül hosztolásra.
2. **Backend**: Azure App Services platformon fut, amely biztosítja a REST API skálázhatóságát és megbízhatóságát.
3. **Adatbázis**: Azure Database for MySQL, amely az ingyenes Azure-tárhelykeretet használja.

**3. Telepítés és követelmények**

**3.1 Hardverkövetelmények**

* **Frontend**:
  + Minimum: Egy ingyenes Azure Static Web App példány.
  + Ajánlott: CDN használat az erőforrások gyorsabb kiszolgálásához.
* **Backend**:
  + Minimum: Egy Azure App Service B1 szintű példány (ingyenes).
  + Ajánlott: Scale-up opció későbbi terhelés esetére.
* **Adatbázis**:
  + Minimum: Azure Database for MySQL (Basic Tier, 5 GB tárhely).
  + Ajánlott: Automatikus skálázási lehetőség aktiválása.

**3.2 Szoftverkövetelmények**

* **Frontend**:
  + Angular CLI (fejlesztéshez).
  + Azure Static Web Apps Deployment Tools.
* **Backend**:
  + .NET Core SDK és Runtime.
  + Entity Framework Core könyvtár.
  + Azure App Service Deployment Tools.
* **Adatbázis**:
  + MySQL Server és kliens szoftver.
  + Azure Database for MySQL konfigurációs eszközök.

**3.3 Azure szolgáltatás-követelmények**

Az ingyenes Azure-kontingens kihasználása érdekében:

* **Azure Free Tier** (ingyenes réteg):
  + 1 db Static Web App.
  + 1 db App Service (Linux vagy Windows alapon).
  + 1 db MySQL Database (5 GB tárolókapacitás).

**3.4 Telepítési folyamat**

1. **Frontend telepítése**:
   * Angular alkalmazás buildelése (ng build --prod).
   * Az alkalmazás deployolása az Azure Static Web App szolgáltatásba.
2. **Backend telepítése**:
   * REST API buildelése és deployolása az Azure App Service-be.  
     dotnet restore   
     dotnet build --configuration Release
   * Visual Studio vagy Azure CLI segítségével történhet meg a deploy
   * Konfigurálás az adatbázis elérésére vonatkozóan.
3. **Adatbázis telepítése**:
   * MySQL adatbázis létrehozása az Azure Database for MySQL-ben.
   * Adatbáziskezelőben dedikált Backend user létrehozása és megfelelő GRANT műveletek kiadása az alkalmazás adatbázisára
   * Kapcsolati string beállítása a backend számára
4. **Verziókezelés**:
   * A Frontend és Backend forráskódot egy Github repository-ban fogjuk fejleszteni.
   * A Github repository main branch-e alapján fog történni minden release.
   * Külön artifactory-t nem fogunk a C# miatt fenntartani.

**Modell adatstruktúra terv**

**Főbb entitások és azok tulajdonságai**

1. **Felhasználó**
   * Felhasználói ID (azonosító)
   * Név, Email, Jelszó (titkosítva)
   * Szerepkör (fogyasztó, termelő, admin)
   * Kedvenc receptek
   * Kedvenc termékek
2. **Recept**
   * Recept ID
   * Cím
   * Leírás
   * Kategória (pl. leves, főétel, desszert)
   * Hozzávalók (kapcsolat a Termék entitással)
   * Lépések
   * Készítési idő
   * Kép URL
   * Recept típus (publikus/privát)
   * Feltöltő (kapcsolat Felhasználóval)
   * Értékelések, Kommentek (kapcsolat a Visszajelzés entitással)
3. **Termék**
   * Termék ID
   * Név
   * Kategória (pl. zöldség, hús)
   * Leírás
   * Ár
   * Raktárkészlet
   * Termelő ID (kapcsolat Felhasználóval)
   * Értékelések, Kommentek
4. **Visszajelzés**
   * Visszajelzés ID
   * Cél (recepthez vagy termékhez kapcsolva)
   * Értékelés (1-5 csillag)
   * Szöveges visszajelzés
   * Felhasználói ID
   * Időbélyeg
5. **Kosár**
   * Kosár ID
   * Felhasználói ID
   * Kosárelemek (kapcsolat a Termékekkel és mennyiséggel)
   * Végösszeg

# Backend végpontok

Bizonyos végpontok után szerepel a „teljes CRUD” megjegyzés (CRUD: create, read, update, delete), ez arra utal, hogy pl. /producer/products végpont esetén létezik

* /producer/products POST végpont egy új termék létrehozására
* /producer/products/:id GET – létező id-vel rendelkező termék lekérésére
* /producer/products/:id PATCH – létező termék módosítására
* /producer/products/:id DELETE - létező termék törlésére

## Fogyasztó

Authentikáció

* /login
* /register

Termékek

* /products – üres paraméter esetén a főoldalon listázandó termékeket adja vissza, de lehetőség lesz szűrésre, rendezésre is (ár/kategória/típus/megbízhatóság/termelő érékelése/termelő területe)
* /products/:id – termék ismertető, leírás, értékelések
* /products/:id/rate – termék értékelése (szöveges, 5/?)
* /products/:id/favourite
* /producers/:id - bemutatkozás, visszajelzések
* /producers/:id/rate - termelő értékelése (szöveges, 5/?)
* /discover - korábbi rendelések/bestseller/legjobb vélemények (új termékek)

Kosár, rendelés

* /cart POST –termék kosárhoz adása
* /cart GET – kosár jelenlegi tartalmának listázása
* /order – rendelés leadása (kosár jelenlegi tartalmára), megadva a szállítási, fizetési adatokat
* /common-orders – felhasználó gyakran rendelt termékei
* /allow-notifications – új akciókról, termékekről szóló értesítések engedélyezése

Feliratkozások

* /subscriptions (teljes CRUD) – feliratkozás egy rendelésre (termékek egy listája) (gyakoriság is paraméter)
* /subscriptions/:id/unsubscribe - ha úgy akar leiratkozni a felhasználó, hogy később vissza tudja állítani (tehát nem törli)
* /subscriptions/:id/resolve - ha bizonyos árucikkek nem elérhetők rendelkezni kell arról, hogy menjen-e a rendelés

Listák

* /list POST – lista létrehozása
* /list/:id/add,remove – termék hozzáadása, levétele
* /list/:id/invite – felhasználó meghívása a lista szerkesztésére
* /list/invitation/:id/accept,deny – lista meghívás elfogadása/elutasítása
* /list/:id/approve - rendelés elfogadása
* /list/:id/order - ha mindenki elfogadta meg lehet rendelni

## Termelő

Authentikáció

* /producer/login
* /producer/register

Profil, termékek

* /producer/profile
* /producer/products teljes CRUD - saját termékek listázása/szerkesztése,stb.
* /producer/package CRUD – több termékből álló csomag összeállítása, szerkesztése

Rendelések

* /producer/orders - jelenleg aktív rendelések, és eladási statisztikák
* /producer/orders/:id/modify,delete
* /producer/orders/:id/ready -rendelés elkészültének jelzése

Értékelések

* /producer/ratings - termelőre és termékekre leadott értékelések megtekintése
* /producer/ratings/:id/react - válasz küldése - lehet külön-külön termék és termelő endpoint

Feliratkozások (fogyasztói feliratkozás a termelő termékeire)

* /producer/subscriptions - feliratkozások megtekintése
* /producer/subscriptions/:id/accept,deny – feliratkozás elutasítása, elfogadása
* /producer/subscriptions/:id

## Adminisztrátor

Authentikáció

* /admin/login

Értékelések kezelése

* /admin/rating/:id/approve
* /admin/rating/:id DELETE
* /admin/producer/:id/profile/approve,disapprove – termelői profil, bemutatkozás elfogadása, elutasítása (elutasítás esetén szöveges magyarázat)
* /admin/product/:id/approve,disapprove
* /admin/package/:id/approve,disapprove

**Project Work Breakdown**

User Story-k az MVP alapján:

1. Fogyasztó regisztráció

2. Termelői regisztráció

3. Termelői nézet – Termékek feltöltése

4. Fogyasztói nézet – Kosár/rendelési lista készítése és elküldése

5. Termelői nézet – Rendelés összekészítése, feladása

6. Termékinformációk és vélemények/visszajelzések

7. Termelői nézet – Profil

8. Fogyasztói nézet – Receptek

**Backend & Adatbázis**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Feladat | User Story | Elő-feltétel | Fejlesztő | Idő (óra) | Elfogadási kritérium |
| A Felhasználó, Recept, Termék, Visszajelzés, és Kosár entitások alapján a szükséges attribútumok, kapcsolatokat és adatbázis szintű konzisztenciát biztosító feltételek meghatározása. | 1-8 |  | Ádám | 6 | Minden entitás tartalmazza az üzleti logikának megfelelő attribútumokat. |
| Az entitások közötti kapcsolatok létrehozása, pl. „Recept - Hozzávalók” több-a-többhöz kapcsolat az adott hozzávaló mennyiségével kiegészítve. | 1-8 |  | Ádám | 4 | Az entitások közötti relációk helyesen meg vannak határozva. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A Felhasználók, Receptek, Termékek, Visszajelzések és Kosarak tábláinak létrehozása az egyes attribútumokkal, valamint a relációk szerinti kapcsolatok implementálása. | 1-4, 6, 8 |  | Ádám | 6 | A feladatban felsorolt táblák léteznek, tartalmazzák az előírt attribútumokat és relációkat. |
| Az entitások közötti kapcsolatok megtervezése, ideértve a külső kulcsokat (pl. Felhasználó ID a Visszajelzés táblában). | 1-8 |  | Ádám | 4 | Az entitások közötti külső kulcsok helyesen hivatkoznak az elsődleges kulcsokra. |
| Az egyes mezők adattípusainak meghatározása az optimalizált tárolás és adatkezelés érdekében, pl. „Ár” mező numerikus típusban, „Leírás” szöveges adattípusként. | 1-8 |  | Ádám | 3 | Minden mező adattípusa illeszkedik az adat által reprezentált információ jellegéhez. |
| Elsődleges kulcsok és idegen kulcsok definiálása az entitások összekapcsolására, pl. a Termelő ID, mint idegen kulcs a Termék táblában. | 1-8 |  | Ádám | 3 | Minden entitás egyedi elsődleges kulccsal rendelkezik, az idegen kulcsok más entitások elsődleges kulcsaira hivatkoznak. |
| Authentikáció végpontok mögötti üzleti logika (Auth) | 1, 2, 6, 8 |  | Kristóf | 5 | A regisztráció és bejelentkezés során a jelszavak titkosítása és tokenek kezelése megfelelően történik. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Fogyasztó - Termékek végpontok mögötti üzleti logika | 4, 6 | Auth | Kristóf | 4 | A fogyasztók számára elérhető a termékek listázása, szűrése, leírása, értékelése … |
| Fogyasztó - Kosár, rendelés végpontok mögötti üzleti logika | 4, 6 | Auth | Kristóf | 4 | A fogyasztó hozzá tud adni terméket a kosarához, meg tudja tekinteni annak tartalmát, le tudja adni az ott lévő termékekre a megrendelést. |
| Fogyasztó - Feliratkozások végpontok mögötti üzleti logika | 4 | Auth | Kristóf | 4 | A fogyasztó fel- és le tud iratkozni rendelésről, nyilatkozni tud termékhiány esetén. |
| Fogyasztó - Listák végpontok mögötti üzleti logika | 4 | Auth | Kristóf | 4 | A fogyasztó létrehozhat, vagy szerkeszthet listákat, akár más felhasználókkal. |
| Termelő - Profil, termékek végpontok mögötti üzleti logika | 3, 7 | Auth | Kristóf | 4 | A termelő fel tud tölteni termékeket és létre tud hozni csomagokat, |
| Termelő - Rendelések végpontok mögötti üzleti logika | 5 | Auth | Kristóf | 4 | A termelő látja meg tudja tekinteni az aktív rendeléseket, szerkesztheti vagy törölheti azokat és jelezheti, ha kész. |
| Termelő - Értékelések végpontok mögötti üzleti logika | 6 | Auth | Kristóf | 4 | A termelő megtekintheti értékeléseit és válaszolhat rájuk. |
| Termelő - Feliratkozások végpontok mögötti üzleti logika | 5, 7 | Auth | Kristóf | 4 | A termelő megtekintheti, elfogadhatja vagy törölheti feliratkozásait. |
| Admin - Értékelések kezelése végpontok mögötti üzleti logika | 6 | Auth | Kristóf | 4 | Az adminisztrátor el tud fogadni vagy utasítani értékeléseket, termelői profilokat. |

**Frontend**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| User Story | Feladat | Feladat leírása | Előfeltétel | Fejlesztő | Idő (óra) | Elfogadási kritérium |
| 1 | 1 | Weboldal „Fogyasztói regisztráció” felület kinézetének megtervezése (wireframe) |  | Gábor | 2 | Wireframe létezik a regisztrációhoz szükséges mezőkkel |
| 1 | 2 | Felület implementálása |  | Gábor | 3 | A regisztrációs oldal interaktív és egyezik az 1.1 wireframe-jével |
| 1 | 3 | Bejelentkezési és regisztrációs lehetőség (wireframe) |  | Gábor | 1 | A felhasználó az 1.2-ben implementált felületet használva képes regisztrálni vagy bejelentkezni |
| 1 | 4 | Bejelentkezés és regisztráció validálása frontend oldalon (karakterszám, érvényes email cím, megfelelő jelszó) |  | Gábor | 2 | A mezők validációja helyesen működik pl. csak érvényes email címmel lehet regisztrálni, bejelentkezni |
| 1 | 5 | Regisztráció elküldése és hitelesítése | backend | Gábor | 1 | A regisztrációs adatokat a backend helyesen feldolgozza, visszajelzést küld |
| 1 | 6 | Bejelentkezés elküldése és hitelesítése | backend | Gábor | 1 | A bejelentkezési adatokat a backend helyesen feldolgozza, JWT tokent küld vissza, ha sikeres |
| 1 | 7 | JWT Token hozzácsatolása a bejelentkezett fogyasztóhoz frontend oldalon (Token interceptor) |  | Gábor | 3 | A token automatikusan csatolásra kerül az API-kérésekhez |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 8 | JWT Token lekérése | backend | Gábor | 1 | A token sikeresen lekérhető |
| 2 | 1 | Weboldal „Termelői regisztráció” felület kinézetének megtervezése |  | Gábor | 2 | Wireframe létezik a regisztrációhoz szükséges mezőkkel |
| 2 | 2 | Felület implementálása |  | Gábor | 3 | A regisztrációs oldal interaktív és egyezik a 2.1 wireframe-jével |
| 2 | 3 | Bejelentkezési és regisztrációs lehetőség (wireframe) |  | Gábor | 1 | A termelő a 2.2-ben implementált felületet használva képes regisztrálni vagy bejelentkezni. |
| 2 | 4 | Bejelentkezés és regisztráció validálása frontend oldalon (karakterszám, érvényes email cím, megfelelő jelszó) |  | Gábor | 2 | A mezők validációja helyesen működik pl. csak érvényes email címmel lehet regisztrálni, bejelentkezni |
| 2 | 5 | Regisztráció elküldése és hitelesítése | backend | Gábor | 1 | A regisztrációs adatokat a backend helyesen feldolgozza, visszajelzést küld |
| 2 | 6 | Bejelentkezés elküldése és hitelesítése | backend | Gábor | 1 | A bejelentkezési adatokat a backend helyesen feldolgozza, JWT tokent küld vissza, ha sikeres |
| 2 | 7 | JWT Token hozzácsatolása a bejelentkezett termelőhöz frontend oldalon (Token interceptor) |  | Gábor | 3 | A token automatikusan csatolásra kerül az API-kérésekhez |
| 2 | 8 | JWT Token lekérése | backend | Gábor | 1 | A token sikeresen lekérhető |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 3 | 1 | Termelői nézetben a „Saját termékek” felület kinézetének megtervezése |  | Gábor | 3 | Wireframe létezik a termelő termékeinek megtekintéséhez |
| 3 | 2 | Gombok létrehozása termékek feltöltésére, módosítására és törlésére |  | Gábor | 3 | A termelő fel tud tölteni új terméket, vagy meglévőt módosítani, törölni |
| 3 | 3 | Felület megtervezése a kiválasztott, vagy feltölteni kívánt termékeknek |  | Gábor | 1 | A kiválasztott vagy feltölteni kívánt termékek részletei megjeleníthetők |
| 3 | 4 | Új termék feltöltésekor adatok bekérése |  | Gábor | 1 | Feltöltéskor kitöltendő űrlap a termék részleteinek megfelelő mezőkkel |
| 3 | 5 | Új termék feltöltésekor az adatok hitelesítése frontend oldalon (hiteles név, darabszám, leírás) |  | Gábor | 2 | A feltöltés csak akkor sikeres, ha minden mező rendeltetésszerűen van kitöltve |
| 3 | 6 | Termék módosításakor az adatok megjelenítése és lehetőség a módosításra |  | Gábor | 1 | A módosítani kívánt termékek részletei megjeleníthetők és a termelő azokat szerkeszteni tudja |
| 3 | 7 | Termék módosításakor az adatok hitelesítése frontend oldalon |  | Gábor | 1 | A módosítás csak akkor sikeres, ha minden mező rendeltetésszerűen van kitöltve |
| 3 | 8 | Új termék feltöltésekor, módosításkor, törléskor backend hitelesítés |  | Gábor | 2 | Az új/módosított/ törölt adatok a backendben helyesen érvényesülnek |
| 3 | 9 | Hitelesítés után a „Saját termékek” oldal dinamikus frissítése |  | Gábor | 1 | Termékek feltöltésekor, módosításakor vagy törlésekor a változások azonal megjelennek az oldalon |
| 4 | 1 | Termékek listázása wireframe |  | Botond | 2 | Wireframe létezik és egyezik a tervezett terméklista kinézetével |
| 4 | 2 | Kosár tartalmának mentése az adott sessionben |  | Botond | 3 | A kosár tartalma helyesen tárolódik és módosul a munkamenet során |
| 4 | 3 | Kosár megtekintése wireframe |  | Botond | 2 | Wireframe létezik és egyezik a tervezett kosártartalom elrendezésével |
| 4 | 4 | Termékek lekérdezése API | backend végpont | Botond | 2 | A termékek és adataik sikeresen lekérdezhetőek |
| 4 | 5 | Kosár termékeinek lekérdezése API | backend végpont | Botond | 2 | A kosár tartalma és a termékek részletei sikeresen lekérdezhetőek |
| 4 | 6 | Termékek listázása nézet tényleges adatokkal | 4.1, 4.4 | Botond | 3 | A termékek listája a helyes termékekkel és részletekkel megjelenik |
| 4 | 7 | Kosár tartalmának listázása tényleges adatokkal | 4.3, 4.5 | Botond | 3 | A kosárban lévő termékek listája a helyes termékekkel és részletekkel megjelenik |
| 4 | 8 | Rendelés leadása gomb -> Backend végpont hívás a rendelés tárolására |  | Botond | 2 | Rendelés leadásakor a rendelés sikeresen mentésre kerül a backendbe |
| 5 | 1 | Összekészítendő rendelések wireframe |  | Botond | 3 | Összekészítendő termelések nézet meg van tervezve |
| 5 | 2 | Rendelés megtekintése wireframe |  | Botond | 2 | Nézet egy adott rendelés megtekintésére meg van tervezve |
| 5 | 3 | Összekészítendő rendelések lekérdezése API | backend végpont | Botond | 3 | Összekészítendő rendelések listája lekérdezhető |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 5 | 4 | Adott rendelés lekérdezése API | backend végpont | Botond | 3 | Adott rendelés adatai lekérdezhetők |
| 5 | 5 | Összekészítendő rendelések nézet tényleges adatokkal | 5.1, 5.3 | Botond | 3 | A termelő meg tudja tekinteni a számára leadott rendelések listáját |
| 5 | 6 | Rendelés megtekintése nézet tényleges adatokkal | 5.2, 5.4 | Botond | 3 | A termelő meg tudja nézni, hogy melyik számára leadott termelés mit és mennyit tartalmaz a termékei közül |
| 5 | 7 | Rendeléshez szállítási információ megadása -> Backend végpont hívása és a rendelés szállításának megkezdése |  | Botond | 2 | A szállítási információk helyesen tárolódnak |
| 6 | 1 | Termék információk wireframe |  | Botond | 3 | Wireframe amely tartalmazza a termék részleteit |
| 6 | 2 | Vélemények komponens (nem külön view) wireframe |  | Botond | 1 | Wireframe amely tartalmazza a létező véleményeket és űrlapot új vélemény írásához |
| 6 | 3 | Termék információk lekérdezése | backend végpont | Botond | 1 | A termékek részletes adatai lekérdezhetők az API-ból |
| 6 | 4 | Vélemények lekérdezése | backend végpont | Botond | 1 | A termékekről írt vélemények lekérdezhetők az API-ból |
| 6 | 5 | Termék információk nézet tényleges adatokkal | 6.1, 6.3 | Botond | 2 | A termékek részletei frontend oldalon helyesen jelennek meg |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 6 | 6 | Vélemények komponens tényleges adatokkal | 6.2, 6.4 | Botond | 3 | A termékekről írt vélemények frontend oldalon helyesen jelennek meg |
| 6 | 7 | Rendelés esetén vélemény írása és leadása -> Backend végpont hívás az értékelés tárolása |  | Botond | 3 | Az írt értékelés sikeresen tárolódik |
| 7 | 1 | Termelői profil wireframe |  | Botond | 2 | Wireframe tartalmazza a termelői adatokat és a termékeket |
| 7 | 2 | Termelői profil adatok szerkesztése -> Backend végpont hívás az adatok tárolására |  | Botond | 4 | A termelő szerkeszteni tudja a profilját |
| 7 | 3 | Termelői profil adatok lekérdezése | backend végpont | Botond | 1 | A termelők adatai sikeresen lekérdezhetők |
| 7 | 4 | Termelői profil nézet tényleges adatokkal | 7.1, 7.2, 7.3 | Botond | 3 | A termelői profil nézet helyesen jeleníti meg a termelő adatait |
| 8 | 1 | Receptek felület kinézetének megtervezése (wireframe mockup) |  | Gábor | 1 | A receptek oldal tartalmazza a létező recepteket |
| 8 | 2 | Lehetőség új receptek feltöltésére |  | Gábor | 2 | A felhasználók képesek új receptet hozzáadni |
| 8 | 3 | A receptek feltöltésekor az adatok bekérése (Leírás, elkészítése, hozzávalók, képek, videók) |  | Gábor | 3 | Új recept hozzáadásakor az űrlap tartalmazza a recept megadásához szükséges mezőket |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 8 | 4 | Az adatok frontend oldali hitelesítése |  | Gábor | 1 | Új recept hozzáadásakor csak a receptek követeményeinek megfelelő adatokat lehet megadni |
| 8 | 5 | Az adatok backend oldali hitelesítése |  | Gábor | 1 | Visszajelzés sikeres receptfeltöltés esetén |
| 8 | 6 | Hitelesítés után a Receptek oldal dinamikus frissítése |  | Gábor | 2 | Új recept feltöltése után az azonal megjelenik a receptek között |
| 8 | 7 | Lehetőség privát és publikus receptek létrehozására |  | Gábor | 2 | A felhasználó be tudja állítani feltöltéskor a recept láthatóságát. Csak a publikus receptek láthatóak mindenki számára |
| 8 | 8 | A hozzávalók alapján előre elkészített csomag megrendelésének a lehetősége |  | Gábor | 2 | Receptek megtekintésekor a felhasználó számára rendelkezésre áll a csomag megrendelésének lehetősége |
| 8 | 9 | Hozzávalókból csomag generálása termelők és termékeik alapján |  | Gábor | 4 | A recept hozzávalói alapján összekészített csomag a megfelelő termékeket tartalmazza |