

Magic Fridge

Projektfeladat specifikáció





Tartalomjegyzék

Tartalom

1	Bevezetés.....	3
1.1.	Program neve	3
1.2.	A feladat rövid ismertetése	3
1.3.	Fő célkitűzések:	3
2	Elvárások a megoldással kapcsolatban	4
2.1.	Operációs rendszer, környezet	4
2.2.	Felhasználandó programozási nyelv	4
2.3.	Megoldás formátuma	4
2.4.	Modulok.....	4
3	Szoftver specifikáció.....	5
3.1.	Megjelenés és főbb UI elemek.....	5
3.2.	Funkciók	5
4	Dokumentáció	7
4.1.	Erőforrás-terv és munkaidő nyilvántartása	7
4.2.	Technikai dokumentáció.....	7
4.3.	Forráskód dokumentáció.....	7
5	A projekt értékelése	8
5.1.	Felhasználói szempontok	8
5.2.	Technikai szempontok	8
5.3.	Piaci értékelés	8
6	Projekt adatlap.....	9



1 Bevezetés

1.1. Program neve

Magic Fridge – Intelligens hűtő- és kamrakezelő rendszer

1.2. A feladat rövid ismertetése

A Magic Fridge olyan alkalmazás, amely lehetővé teszi a felhasználók számára, hogy nyomon kövessék otthoni hűtőjük és kamrájuk tartalmát. A rendszer támogatja az élelmiszerek lejáratí idejének figyelését, valamint képes automatikus bevásárlólistát készíteni az elfogyott vagy hiányzó termékek alapján. A felhasználók az aktuális készletük szerint receptajánlásokat is kaphatnak, külső API (pl. Spoonacular) segítségével. A cél az élelmiszerpazarlás csökkentése, a vásárlás megkönnyítése, valamint a főzési ötletek bővítése.

1.3. Fő célkitűzések:

- Biztosítsa a hűtő és a kamra tartalmának naprakész nyomon követésére, beleértve a mennyiségeket, kategóriákat és lejáratí időkét.
- Az élelmiszerpazarlás csökkentése azáltal, hogy a rendszer figyelmeztet a közeli lejáratí idejű termékekre.
- A bevásárlás megkönnyítése egy olyan funkcióval, amely automatikusan javasol bevásárlólistát az aktuális készlet alapján.
- A felhasználók számára receptajánlatok biztosítása a már meglévő alapanyagok alapján, ezáltal segítve a változatos ételkészítést.
- Többfelhasználós adatkezelés megvalósítása, amely lehetővé teszi, hogy egy háztartás tagjai közösen kezeljék a készleteiket és bevásárlólistáikat.



2 Elvárások a megoldással kapcsolatban

2.1. Operációs rendszer, környezet

- Windows 10, 11
- Android, IOS
- Backend környezet: ASP.NET Core (C#).
- Frontend: HTML, CSS, JavaScript.
- Adatbázis: MySQL
- Külső API: Spoonacular.

2.2. Felhasználandó programozási nyelv

- Backend: C# (ASP.NET Core).
- Frontend: JavaScript / HTML / CSS.
- Kiegészítő fejlesztéshez: opcionálisan Python szkriptek.

2.3. Megoldás formátuma

- Teljes projekt forráskód ,backend és frontend.
- Verziókezelő repository, Google Drive illetve GitHub.
- Adatbázis export és adatmodell diagram.
- Felhasználói dokumentáció PDF formátumban és képernyő képekkel.
- Tesztelési jegyzőkönyv illetve eredmények.

2.4. Modulok

- Adatbázis tervezés és implementáció (hűtő, kamra, bevásárlólista, felhasználói fiókok)
- Felhasználói fiók kezelés (regisztráció, autentikáció, jogosultságok)
- Hűtő- és kamra tartalom kezelése (hozzáadás, szerkesztés, törlés)
- Automatikus és manuális bevásárlólista készítés
- Külső API integráció recept ajánláshoz
- Mobiltelefonos beolvasás és készletfrissítés
- Értesítési rendszer (lejárat, hiányzó termék, új lista)
- Statisztikák (kidobott termékek, pazarlás).



3 Szoftver specifikáció

3.1. Megjelenés és főbb UI elemek

- A Magic Fridge felhasználói felülete letisztult, modern és reszponzív kialakítású, amely a különböző eszközökön (asztali böngésző, mobiltelefon, tablet) egyaránt jól használható. A frontend HTML, CSS és JavaScript felhasználásával készül, Bootstrap vagy más modern CSS-keretrendszer alkalmazásával.
- **Főmenü:** Könnyen elérhető navigációs sáv, amely lehetővé teszi a gyors váltást a modulok között (Hűtő, Kamra, Bevásárlólista, Receptek, Statisztikák, Profil).
- **Dashboard (kezdőoldal):** Áttekintő nézet a készletekről, lejáró termékekről, és friss receptajánlatokról.
- **Táblázatos nézetek:**
 - **Hűtő és kamra tartalma:** Termékek neve, kategóriája, lejárat dátuma, mennyiség, hozzáadás ideje.
 - **Szerkesztési lehetőségek:** Minden sorhoz "Szerkesztés", "Törlés" gomb, inline módosítás opciók.
- **Formok:** Egyszerűen kitölthető űrlapok új termék hozzáadásához, bevásárlólista szerkesztéséhez.
- **Reszponzivitás:** A felület mobilbarát, a fontos funkciók mobilnézetben is könnyen elérhetőek (pl. "plusz" gomb úszó ikonként új termék hozzáadásához).

3.2. Funkciók

1. Felhasználói autentikáció:

- Regisztráció (e-mail + jelszó), e-mail validáció
- Bejelentkezés biztonságos jelszókezeléssel (ASP.NET Identity)
- Jogosultságkezelés: külön háztartási egységek kezelése, adminisztrátor illetve felhasználó

2. Hűtő és kamra tartalom kezelése

- Új termékek hozzáadása kategória, név, mennyiség, lejárat dátum, helyszín (hűtő/kamra) szerint
- Termékek szerkesztése, törlése
- Termékek mennyiségének frissítése vásárlás vagy felhasználás után



3. Automatikus és kézi bevásárlólista kezelés

- **Automatikus generálás:** elfogyott vagy alacsony mennyiségű termékek alapján
- **Kézi lista létrehozás:** felhasználó manuálisan adhat hozzá elemeket
- **Közös lista:** háztartás tagjai közösen szerkeszthetik, szinkronizált állapotban
- Lista exportálása (PDF vagy nyomtatható nézet)

4. Receptajánlás (Spoonacular API integrációval)

- API hívás a rendelkezésre álló alapanyagok alapján
- Szűrés: kalória, ételtípus, diéta (pl. vegetáriánus, gluténmentes)
- Részletes receptoldal: hozzávalók, elkészítési lépések, előkészületi idő, tápanyag-információk
- Recept mentése "Kedvencek" közé

5. Vonalkód-/QR-kód beolvasás (mobilos funkció)

- Mobilkamera használata vonalkód beolvasására
- Termék automatikus felismerése adatbázisból vagy kézi adatbevitel

6. Értesítési rendszer

- Bevásárlólista változások, új receptjavaslatok
- Testre szabható értesítési beállítások a felhasználói fiókban



4 Dokumentáció

4.1. Erőforrás-terv és munkaidő nyilvántartása

- A fejlesztési feladatok (adatbázis-tervezés, backend, frontend, API-integráció, dokumentáció) csapattagok között kerülnek szétosztásra. Munkaidő-nyilvántartás készül, amely a projekt végén elemzésre kerül.

4.2. Technikai dokumentáció

- Tartalmazza a rendszer architektúráját, az adatbázis sémát, az API-végpontokat, a Spoonacular integráció leírását, valamint a fő funkciók működési logikáját.

4.3. Felhasználói dokumentáció

- Útmutató képernyőképekkel, amely bemutatja a regisztrációt, a hűtő és kamra kezelést, a bevásárlólista használatát, a receptajánló működését, valamint a mobiltelefonos beolvasást.

4.4. Forráskód dokumentáció

- Ez a rész segíti más fejlesztők vagy a későbbi karbantartók munkáját, hogy megértsék a kód működését.
- **Kód struktúrája:** A főbb mappák és fájlok rövid bemutatása
- **Kommentárok és funkcióleírások:** A forráskódban szereplő függvényekhez, osztályokhoz írt rövid magyarázatok.
- **Útmutató képernyőképekkel:**
 - Bemutatja a rendszer főbb funkcióit képi példákon keresztül:
 - Hogyan történik a felhasználói regisztráció és bejelentkezés
 - Miként lehet élelmiszereket hozzáadni a hűtő/kamra készlethez
 - Hogyan használható a bevásárlólista funkció
 - Milyen módon jelennek meg a receptajánlások
 - Hogyan működik a mobiltelefonos vonalkód-beolvasás



5 A projekt értékelése

5.1. Felhasználói szempontok

- Könnyen kezelhető, áttekinthető felület
- Reszponzív működés minden eszközön
- Megbízható bevásárlólista és receptajánló funkció
- Értesítések pontos és időben történő megjelenése
- Igényes felhasználói dokumentáció

5.2. Technikai szempontok

- Tiszta, dokumentált forráskód
- Kódolási konvenciók betartása
- Stabil működés, sikeres tesztelések
- Verziókezelt fejlesztés (GitHub)
- Teljes körű dokumentáció

5.3. Piaci értékelés

- Élelmiszerpazarlás csökkentése, környezettudatosság támogatása
- Többfelhasználós háztartások számára kiemelt érték
- Lehetséges kiterjesztés okoseszköz-integrációra (pl. okoshűtők)
- Kockázatok: adatbiztonság, felhasználói elfogadás, API-használati korlátok



6 Projekt adatlap

- **Projekt neve:** Magic Fridge
- **Projekt rövid ismertetése:** Intelligens élelmiszerkészlet-kezelő rendszer, amely nyomon követi a hűtő és kamra tartalmát, automatikusan készít bevásárlólistát, receptajánlókat kínál külső API alapján, és értesít a lejáratáról.
- **Felhasznált programozási nyelvek:** C#, JavaScript, HTML, CSS
- **Adatbázis:** MySQL
- **Külső API:** Spoonacular
- **Készítette:** Sutús Róbert, Udvari Dominik