**Fejlesztői dokumentáció**

**Tartalomjegyzék**

1. **Bevezető**
2. **Feladatkörök  
   Iroczki Béla, Vályi Roland Gábor, Radácsi Dávid - Frontend fejlesztés, Backend fejlesztés és API-k, Adatbázis és MySQL kezelés**
3. **Fejlesztői környezet bemutatása  
   2.1. Visual Studio 2022  
   2.2. Visual Studio Code  
   2.3. MySQL - phpMyAdmin**
4. **Használt technológiák  
   3.1. ReactJS  
   3.2. ASP.NET Core Web API  
   3.3. GitHub  
   3.4. Bootstrap CSS - FontAwesome Icons  
   3.5. MySQL  
   3.6. NPM**
5. **A Program  
   4.1. A Program leírása  
   4.2. Deploy platform és Publikáció  
   4.2.1. Linux otthoni szerver  
   4.2.2. Cloudflare - Zero Trust Tunnel és Domain**
6. **Tájékozódás az oldalon belül  
   5.1. Főoldal  
   5.2. BMI kalkulátor  
   5.3. Étrendajánló  
   5.4. Kalóriaszámoló  
   5.5. Saját Profil  
   5.6. Admin panel  
   5.7. Bejelentkezés  
   5.8. Regisztráció  
   5.9. Kijelentkezés**
7. **Backend**

**6.1. Authorization végpontok**

**6.2. Backend végpontok**

**6.3. Adatkezelés és biztonság**

1. **Forrásjegyzék  
   7.1. Visual Studio 2022  
   7.2. ASP.NET Core Web API  
   7.3. ReactJS  
   7.4. Word  
   7.5. NPM  
   7.6. Bootstrap**
2. **Forrásjegyzék**
3. **Bevezető**

**Napjainkban az egészséges életmódra való törekvés egyre fontosabbá válik, hiszen a helytelen táplálkozás és a mozgásszegény életmód számos egészségügyi problémát okozhat. Az egészségtudatos életmód kialakításához azonban nem mindig könnyű megtalálni a megfelelő eszközöket és információkat. Projektünk célja egy olyan könnyen használható webes alkalmazás fejlesztése, amely támogatja a felhasználókat az egészségük megőrzésében és a kitűzött céljaik elérésében.**

**1. Feladatkörök**

**A projekt megvalósításában részt vevő csapattagok szerepkörét az alábbiakban részletezzük:**

**1.1. Backend fejlesztő**

**A backend fejlesztő felelős az alkalmazás szerveroldali logikájáért, az adatbázis tervezéséért és kezeléséért, valamint az API végpontok implementálásáért.**

**Főbb feladatai:**

**Az adatbázis struktúrájának kialakítása MySQL segítségével.**

**Az API végpontok megtervezése és implementálása Node.js segítségével.**

**Adatbiztonsági megoldások, például jelszó-hashelés, token-alapú hitelesítés.**

**Hibakezelés és teljesítmény-optimalizálás a szerveroldalon.**

**1.2. Frontend fejlesztő**

**A frontend fejlesztő felelős a felhasználói felület kialakításáért, a vizuális megjelenésért, valamint az interakciók működéséért.**

**Főbb feladatai:**

**A React komponensek megírása a felhasználói felülethez.**

**Reszponzív dizájn készítése Bootstrap és egyedi CSS segítségével.**

**API-k hívása és az adatok dinamikus megjelenítése a felhasználóknak.**

**Felhasználói élmény (UX) optimalizálása, könnyen kezelhető navigáció biztosítása.**

**1.3. Projektmenedzser**

**A projektmenedzser koordinálja a csapattagok munkáját, felügyeli a projekt előrehaladását, és gondoskodik arról, hogy a célkitűzések időben teljesüljenek.**

**Főbb feladatai:**

**A projekt céljainak és követelményeinek meghatározása.**

**Határidők betartásának ellenőrzése, feladatok priorizálása.**

**Kommunikáció a csapattagok és az esetleges külső érintettek között.**

**2. Fejlesztői környezet bemutatása**

**A projekt megvalósítása során a következő fejlesztői környezeteket és eszközöket használtuk:**

**2.1. Visual Studio Code**

**A Visual Studio Code a frontend és backend fejlesztés fő eszköze.**

**Funkciók, amelyeket használunk:**

**Beépített terminál Node.js parancsok futtatására.**

**Kódformázás és hibakeresés.**

**Git integráció a verziókezeléshez.**

**2.2. Node.js környezet**

**A Node.js lehetővé teszi a backend gyors és megbízható fejlesztését.**

**Fontos modulok:**

**Express.js: A REST API végpontok létrehozásához.**

**jsonwebtoken: Hitelesítési megoldások tokenekkel.**

**bcryptjs: Jelszavak titkosítására és ellenőrzésére.**

**2.3. MySQL és phpMyAdmin**

**A MySQL szolgáltatja az adatbázis-kezelést, míg a phpMyAdmin vizuális eszközként segít az adatok kezelésében.**

**2.4. GitHub**

**A verziókezelésre a GitHub szolgál.**

**Használat:**

**A kód tárolása és megosztása a csapat tagjai között.**

**Pull requestek és issue-k kezelésére.**

**3. Használt technológiák**

**3.1. ReactJS**

**A ReactJS keretrendszer szolgáltatja a frontend alapjait. A komponensalapú megközelítés lehetővé teszi az újrafelhasználható elemek egyszerű kezelését.**

**3.2. Node.js és Express.js**

**A szerveroldali fejlesztés során az Express.js keretrendszert használtuk a REST API-k kezelésére.**

**3.3. Bootstrap és CSS**

**A vizuális megjelenés Bootstrap alapon készült, amelyhez egyedi CSS-t is használtunk az arculat testreszabására.**

**3.4. FontAwesome ikonok**

**A weboldalon használt ikonok a FontAwesome könyvtárból származnak, amelyek vizuálisan kiemelik a navigációs elemeket és funkciókat.**

**4. A Program**

****

**4.1. Alapfunkciók**

**A program három fő pilléren nyugszik:**

1. **BMI-számítás: A felhasználók megadják a testsúlyukat és magasságukat, az alkalmazás pedig kiszámítja a testtömegindexüket.**
2. **Étrendajánlás: Személyre szabott javaslatokat adunk a napi kalóriabevitelhez és tápanyagok arányához.**
3. **Kalóriaszámítás: A felhasználók napi aktivitási szintje alapján kiszámítjuk, hogy mennyi kalóriát kell elfogyasztaniuk céljaik eléréséhez.**

**4.2. Backend infrastruktúra**

**A program mögött egy REST API húzódik, amely az adatbázisból kéri le az adatokat és biztosítja a kommunikációt a frontend és backend között.**

**5. Tájékozódás az oldalon belül**

**A weboldal navigációs rendszerét úgy alakítottuk ki, hogy könnyen átlátható és intuitív legyen:**

**Főoldal: Áttekintést nyújt az alkalmazásról.**

**BMI-kalkulátor: Azonnali eredményt ad a felhasználónak.**

**Profil: A felhasználó itt kezelheti az adatait és követheti céljait.**

**6. Backend**

A backend a projekt egyik legfontosabb része, mivel az adatokat és a felhasználói interakciókat kezeli. A backend Node.js keretrendszerben készült, és egy MySQL adatbázis tárolja az összes adatot.

**6.1. Authorization végpontok**

A backend rész biztosítja a felhasználói hitelesítést és jogosultságkezelést. Az alábbi végpontok felelnek a felhasználói adatok kezeléséért és a rendszerhez való hozzáférés biztosításáért.

**register**

A felhasználók regisztrálása az alkalmazásba, az alapvető adatok (név, e-mail, jelszó) megadásával.

**login**  
A felhasználók bejelentkezése, amely lehetővé teszi számukra a személyre szabott szolgáltatások, mint a BMI kalkulátor és étrendajánló elérését.

**logout**  
A felhasználó kijelentkezése az alkalmazásból.

**change-username**  
Lehetőség a felhasználónév módosítására, ha szükséges.

**change-password**  
Jelszó módosítása, ha a felhasználó elfelejtette vagy szeretné megváltoztatni a jelszavát.

**send-reset-email**  
A jelszó visszaállításához szükséges e-mail küldése a felhasználónak.

**reset-password**  
A jelszó visszaállítása a kapott reset kóddal.

* **is-admin**  
  A rendszergazdai jogosultságok ellenőrzése, melyek lehetővé teszik az admin panel használatát.

**6.2. Backend végpontok**

A backend végpontok az alkalmazás funkcióit biztosítják, mint például a felhasználói adatok, kalóriaszámolás és étrendajánlás. Ezek a végpontok kezelik a szükséges adatokat és biztosítják a megfelelő válaszokat az alkalmazás számára.

**/api/user/{id}/bmi**  
A felhasználó BMI-jének kiszámítása és visszaadása a felhasználó aktuális súlyának és magasságának figyelembevételével.

**/api/user/{id}/calories**  
A napi kalóriaszükséglet meghatározása a felhasználó célja (tömegnövelés, fogyás vagy szinten tartás) és aktivitási szintje alapján.

**/api/user/{id}/diet**  
Személyre szabott étrend ajánlása a felhasználó céljainak megfelelően, figyelembe véve a napi kalóriaszükségletet és a makrotápanyagok eloszlását (fehérje, szénhidrát, zsír).

**/api/user/{id}/goals**  
A felhasználói célok (pl. súlycsökkentés, izomtömeg növelése) rögzítése és kezelése, amelyek alapján az alkalmazás személyre szabott étrendet és edzéstervet ajánlhat.

**/api/user/{id}/profile**  
A felhasználó adatainak (név, e-mail, súly, magasság, aktivitási szint) frissítése és kezelése.

**/api/user/{id}/activity**  
A felhasználó napi aktivitási szintjének rögzítése, amely segíti a napi kalóriabevitel és a szükséges makrotápanyagok pontos meghatározását.

**/api/admin/users**  
Az adminisztrátori felhasználói felület, amely lehetővé teszi a felhasználói adatok kezelését és módosítását.

**/api/admin/statistics**  
A felhasználói aktivitás és a különböző célok elérésének statisztikai elemzése, amely segíti a fejlesztőket és adminokat az alkalmazás fejlesztésében.

**6.3. Adatkezelés és biztonság**

A backend a felhasználói adatokat biztonságosan tárolja. A jelszavak titkosítva kerülnek mentésre a legmodernebb algoritmusok használatával. Az API-k csak a megfelelő jogosultságú felhasználók számára elérhetőek, és minden adatforgalom titkosítva történik SSL protokollal.

**7. Forrásjegyzék**

A projekt során felhasznált eszközöket és technológiákat az alábbiakban soroljuk fel. Mindezek hozzájárultak a projekt sikeres megvalósításához.

**7.1. Visual Studio 2022**

A backend fejlesztéséhez használt eszköz, amely az .NET Core és Express.js fejlesztésére is támogatást nyújtott.

**7.2. ASP.NET Core Web API**

A backend szolgáltatásai, beleértve az adatkezelést és az API végpontokat, ASP.NET Core Web API keretrendszerrel készültek.

**7.3. ReactJS**

A frontend megvalósításához használt keretrendszer, amely lehetővé tette a dinamikus, komponens alapú felhasználói felület készítését.

**7.4. Word**

A dokumentációk és más szöveges anyagok készítéséhez használt eszköz.

**7.5. NPM**

A Node.js csomagkezelő eszköze, amely segített a szükséges csomagok, például a React telepítésében.

**7.6. Bootstrap**

A vizuális megjelenéshez használt CSS framework, amely segített a reszponzív design kialakításában, illetve elősegítette a gyors fejlesztést a kész komponensek felhasználásával.

**8. Forrásjegyzék**

1. Wikipedia contributors. "React (JavaScript)." Wikipedia, The Free Encyclopedia. [<https://en.wikipedia.org/wiki/React_(JavaScript)>]
2. Wikipedia contributors. "Node.js." Wikipedia, The Free Encyclopedia. [<https://en.wikipedia.org/wiki/Node.js>]
3. Microsoft Docs. "Visual Studio 2022." Microsoft. [<https://learn.microsoft.com/en-us/visualstudio/>]
4. Bootstrap. "Getting Started with Bootstrap." [https://getbootstrap.com/docs/5.0/getting-started/introduction/]
5. MySQL Documentation. "MySQL Server." Oracle. [<https://dev.mysql.com/doc/>]
6. Node.js Documentation. "Node.js." [https://nodejs.org/en/docs/]
7. GitHub Docs. "GitHub: A Guide." GitHub. [<https://docs.github.com/en/github>]