BGSzC Pestszentlőrinci Közgazdasági és Informatikai Szakgimnáziuma

1184 Budapest Hengersor 34.

1. Záró dolgozat

Személyi Edző Kereső

Konzulens tanár: Dobrocsi Csilla Készítette: Chen Wen Jie, Pribelszki Dávid

Tartalom

[1 Bevezetés 3](#_Toc85723173)

[1.1 Feladat leírás 3](#_Toc85723174)

[1.2 A felhasznált ismeretek 3](#_Toc85723175)

[1.3 A felhasznált szoftverek 3](#_Toc85723176)

[2 Felhasználói dokumentáció 4](#_Toc85723177)

[2.1 A program általános specifikációja 4](#_Toc85723178)

[2.2 Rendszerkövetelmények 4](#_Toc85723179)

[2.2.1 Hardver követelmények 4](#_Toc85723180)

[2.2.2 Szoftver követelmények 4](#_Toc85723181)

[2.3 3. A program telepítése 4](#_Toc85723182)

[2.4 A program használatának a részletes leírása 5](#_Toc85723183)

[3 Fejlesztői dokumentáció 6](#_Toc85723184)

[3.1 Az alkalmazott fejlesztői eszközök 6](#_Toc85723185)

[3.2 Adatmodell leírása 6](#_Toc85723186)

[3.3 Részletes feladatspecifikáció, algoritmusok 6](#_Toc85723187)

[3.4 Tesztelési dokumentáció 7](#_Toc85723188)

[4 Összefoglalás 8](#_Toc85723189)

[4.1 Önértékelés 8](#_Toc85723190)

[4.2 Továbbfejlesztési lehetőségek 8](#_Toc85723191)

[5 Felhasznált irodalom 9](#_Toc85723192)

[6 Ábrajegyzék 10](#_Toc85723193)

# Bevezetés

## Feladat leírás

Szakdolgozatunk keretében egy "FitConnect" nevű webalkalmazást fejlesztettünk, amely hidat képez a személyi edzők és a potenciális ügyfelek között. A projekt célja egy olyan platform létrehozása volt, ahol az edzők hatékonyan bemutathatják szolgáltatásaikat, az ügyfelek pedig egyszerűen megtalálhatják a számukra legmegfelelőbb edzőt, kapcsolatba léphetnek velük emailen keresztül, és időpontot foglalhatnak egy integrált rendszer segítségével. A fejlesztést ketten végeztük, megosztva a frontend, backend és dizájn feladatköröket, hogy egy modern, reszponzív és felhasználóbarát alkalmazást hozzunk létre.

A "FitConnect" egy olyan platform, amely két fő célcsoportot szolgál ki: a személyi edzőket és az ügyfeleket. Az edzők részletes profilt hozhatnak létre, amely tartalmazza szolgáltatásaikat, áraikat, elérhetőségüket és szakmai hátterüket. Az ügyfelek szűrők segítségével kereshetnek edzőket (pl. helyszín, szakterület, ár), megtekinthetik profiljaikat, és időpontot foglalhatnak egy beépített naptárrendszeren keresztül. A platform egyszerű és intuitív kezelést biztosít mindkét fél számára, támogatva a hatékony kommunikációt és az edzéstervezést.

## A felhasznált ismeretek

A projekt során a következő területeken szerzett ismereteinket alkalmaztuk:

* **Frontend fejlesztés**: Modern JavaScript/TypeScript, React komponensek fejlesztése.
* **UI/UX tervezés**: Reszponzív felületek, Tailwind CSS és Shadcn/ui használata.
* **Backend alapok**: Adatmodellezés, API integráció (pl. email értesítések).
* **Verziókezelés**: Git és GitHub közös munkafolyamatokhoz.
* **Projektmenedzsment**: Feladatok megosztása és koordináció két fejlesztő között.

A munkát megosztottuk: Pribelszki Dávid a frontend fejlesztésre és dizájnra fókuszált, míg Chen Wen Jie az adatmodellezést és a funkcionális logikát támogatta.

## A felhasznált szoftverek

A fejlesztés során a következő technológiákat és eszközöket használtuk:

* **React 18.3.1**: Gyors és modern frontend keretrendszer, hatékony komponenskezeléshez.
* **TypeScript**: Erősen típusos programozás a hibák csökkentésére és jobb fejlesztői élményre.
* **Vite**: Gyors build eszköz és fejlesztői szerver a hatékony munkához.
* **Tailwind CSS**: Utility-first CSS keretrendszer a dizájn gyorsítására.
* **Shadcn/ui**: Könnyen integrálható React komponens könyvtár az egységes UI-hoz.
* **Lucide-react**: Modern ikonok az esztétikus megjelenésért.
* **Node.js és npm**: Futtatókörnyezet és csomagkezelés.
* **PostgreSQL**: Adatbázis az edzői és foglalási adatok tárolására.
* **Git**: Verziókezelés a közös munkához.

# Felhasználói dokumentáció

## A program általános specifikációja

Az alkalmazás négy fő modulból áll:

1. **Landing oldal**: Bemutatja a platformot, kiemeli előnyeit, és gyors keresést kínál.
2. **Edzők kezelése**: Edzői profilok, keresési és szűrési funkciók (helyszín, szakterület, ár).
3. **Foglalási rendszer**: Naptár alapú időpontfoglalás emailes visszaigazolással.
4. **Felhasználói fiók**: Edzők és ügyfelek számára külön kezelőfelület.

## Rendszerkövetelmények

### Hardver követelmények

* **Hardver**: Minimum 4 GB RAM, 2 GHz processzor, stabil internetkapcsolat.
* **Szoftver**: Modern böngésző (Chrome, Firefox, Edge), Node.js 18+ a fejlesztéshez.

### Szoftver követelmények

## 3. A program telepítése

A fejlesztői környezet beállítása:

* Repository klónozása: git clone <repository-url>
* Függőségek telepítése: npm install
* Fejlesztői szerver indítása: npm run dev
* 

1. kép Ez a módosító ablak a programban

Ajánlott terjedelem: 2 -4 oldal, ábrákkal együtt.

## A program használatának a részletes leírása

Az alkalmazás főbb funkciói és használatuk:

1. **Bejelentkezés/Regisztráció**

* A regisztráció két fiókkal történhet:
  + Felhasználó
  + Edző

1. **Edzők keresése**:
   * Az ügyfelek a főoldalon vagy a keresési oldalon szűrhetik az edzőket (pl. "Budapest, jóga, 5000 Ft/óra").
   * Az edzői profil részletes adatokat mutat (név, tapasztalat, árak), és lehetőséget ad emailes kapcsolatfelvételre vagy foglalásra.
   * Funkció: Szűrők alkalmazása, profil megtekintése, "Foglalás" gomb.
2. **Időpontfoglalás**:
   * Az edző profilján a naptárban láthatóak a szabad időpontok.
   * Az ügyfél kiválaszt egy időpontot, majd a rendszer emailben értesíti mindkét felet.

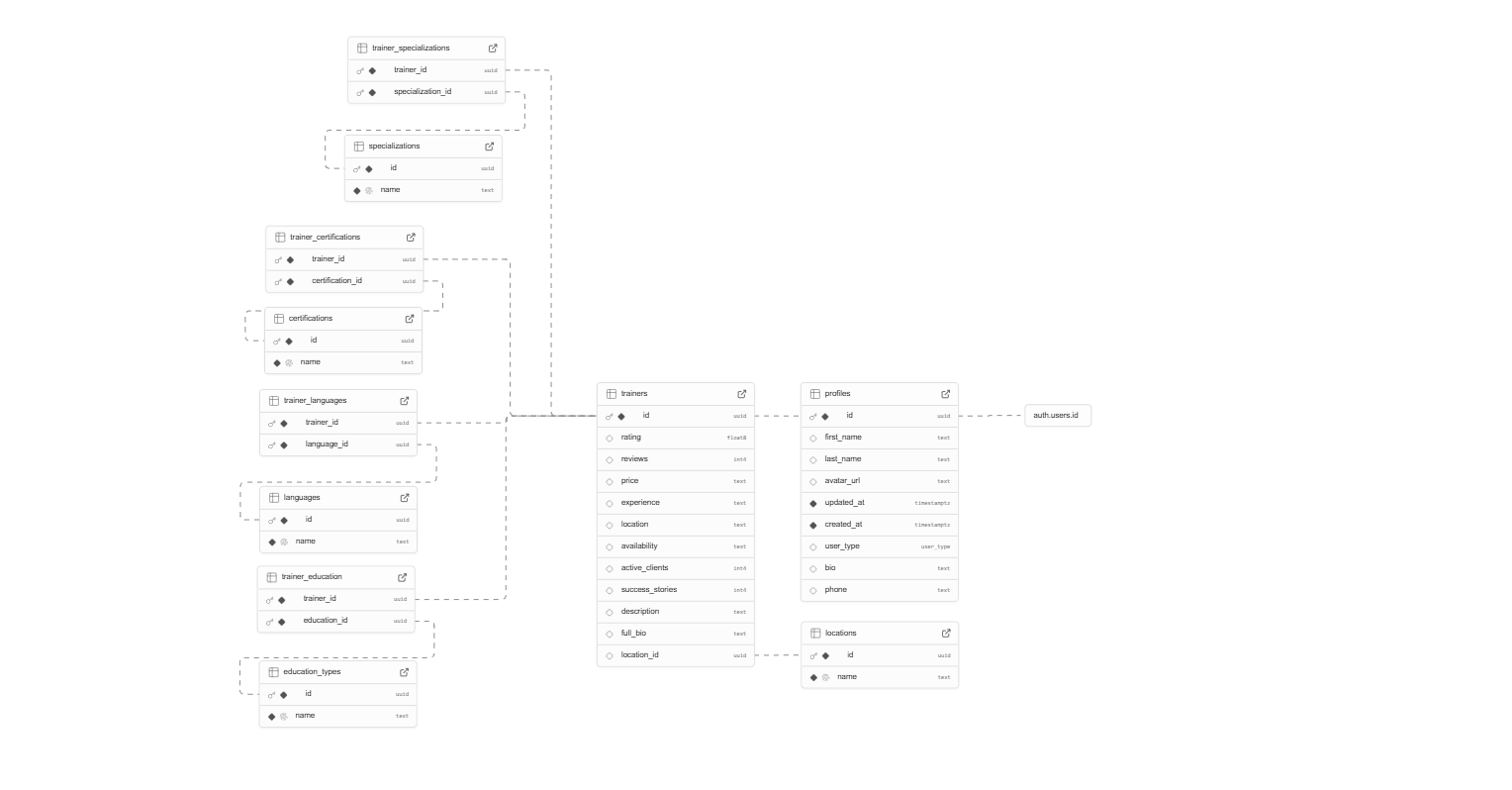
# Fejlesztői dokumentáció

## Az alkalmazott fejlesztői eszközök

A fejlesztés során használt eszközök és környezetek:

* - Visual Studio Code - Elsődleges kódszerkesztő
* - Git/GitHub: Verziókezelés és kollaboráció.
* - Node.js - JavaScript futtatókörnyezet
* - npm - Csomagkezelő
* - Chrome DevTools - Fejlesztői eszközök
* Supabase - Adatbázis kezelés [Supabase | The Open Source Firebase Alternative](https://supabase.com/)
* Shadcn-ui – Hogy valamivel rugalmasabb dizájn biztosítsunk helyenként [Build your component library - shadcn/ui](https://ui.shadcn.com/)
* Tailwind-CSS - [Tailwind CSS - Rapidly build modern websites without ever leaving your HTML.](https://tailwindcss.com/)
* React
* TypeScript
* Vite
* Lucide-react - Kevésbé rugalmas, de egységes dizájnt biztosít.

## Adatmodell leírása !



## Részletes feladatspecifikáció, algoritmusok

A fejlesztés során implementált főbb komponensek és funkciók:

1. Navigációs rendszer:

- React Router-rel kezelt útvonalak (pl. /coaches/:id az edzői profilhoz).

- Dinamikus navigáció

- Védett útvonalak a bejelentkezéshez kötött funkciókhoz.

2. Komponens architektúra:

- Újrafelhasználható komponensek (pl. CoachCard, BookingForm).

- Shadcn/ui integráció a konzisztens dizájnért.

- Egyedi hook-ok fejlesztése

3. Állapotkezelés:

- React hooks használata

- Lokális komponens állapotok

- Globális alkalmazás állapot

4. UI/UX megvalósítás:

- Tailwind CSS alapú styling

- Reszponzív dizájn

- Animációk és átmenetek

## 5. Foglalási logika:

- Algoritmus: Az edző availability tömbjéből szűri a szabad időpontokat, majd a Booking objektumot menti az adatbázisba.

## Tesztelési dokumentáció !

A fejlesztés során alkalmazott tesztelési módszerek:

1. Komponens tesztek:

2. UI tesztek:

- Reszponzivitás ellenőrzése

- Különböző képernyőméretek tesztelése

- Böngésző kompatibilitás

3. Teljesítmény tesztek:

- Betöltési idők optimalizálása

- Komponens renderelés

- Memória használat

# Összefoglalás

## Önértékelés !

Sikeresen létrehoztunk egy modern webalkalmazást, amely:

* Reszponzív és felhasználóbarát felületet és újrafelhasználható komponens struktúrát kínál.
* Hatékonyan kezeli az edző-ügyfél kapcsolatokat.
* Stabil technológiai alapokra épül.

A csapaton belüli munkamegosztás gördülékeny volt, a kommunikáció és a Git használata segítette az együttműködést.

## Továbbfejlesztési lehetőségek

* Fizetési rendszer integrálása (pl. Stripe)
* Mobilalkalmazás fejlesztése

# Felhasznált irodalom !

1. React dokumentáció (<https://reactjs.org/docs>)
2. TypeScript kézikönyv (<https://www.typescriptlang.org/docs>)
3. Tailwind CSS dokumentáció (<https://tailwindcss.com/docs>)
4. Web Development Best Practices - MDN ([https://developer.mozilla.org](https://developer.mozilla.org/))
5. Modern JavaScript Tutorial (<https://javascript.info>)
6. Youtube tutorial videók [YouTube](https://www.youtube.com/)

# Ábrajegyzék

[1. kép Ez a módosító ablak a programban 5](#_Toc63250394)