BrainBoost

# Weboldali Frontend

Tartalom

[Weboldali Frontend 1](#_Toc195711944)

[Fejlesztési napló 1](#_Toc195711945)

[1. Bevezetés 1](#_Toc195711946)

[2. Főbb Funkcionalitások és Fejlesztések 1](#_Toc195711947)

[3. Tesztelési Terv 2](#_Toc195711948)

[Hitelesítés 2](#_Toc195711949)

[Feladatkezelés 3](#_Toc195711950)

[UI és navigáció 3](#_Toc195711951)

[Automatizált tesztek 3](#_Toc195711952)

[Tesztelési eredmények 4](#_Toc195711953)

## Fejlesztési napló

Készült: 2025.04.14. – 2025.04.16.

Fejlesztő: Schlesser Benedek, Plavecz Barnabás

## 1. Bevezetés

Ez a dokumentáció a BrainBoost weboldal fejlesztési folyamatát írja le. A rendszer célja, hogy egyszerű és hatékony módot biztosítson a tanárok és diákok közötti feladatkezelésre, értékelésre, visszajelzésre, illetve fájlok megosztására.

## 2. Főbb Funkcionalitások és Fejlesztések

* 2025.01.24

1. vite+npm: A projekt alapjainak létrehozása Vite és npm segítségével.
2. login/register skeleton: Bejelentkezési és regisztrációs oldal kezdeti verziója.

* 2025.01.25–2025.02.10

1. Regisztráció stílus javítás (01.25)
2. Tanár oldal hozzáadása (02.01): TeacherPage implementálása.
3. Teljes logout funkció (02.10)
4. Felhasználónév megjelenítése profilnál

* 2025.02.17–02.25

1. Diák bejelentkezés teljesítése
2. Navigációs linkek javítása
3. Regisztráció és bejelentkezés angolról magyarra fordítása
4. Tantárgyválasztás és felhasználói regisztrációs folyamat lezárása
5. Diák regisztrációs gomb hozzáadása

* 2025.03.01–03.18

1. Tantárgyak szűrése tanárok szerint (03.01)
2. Dinamikus aktív navigációs sáv (03.09)
3. API-k átalakítása tanár/diák hívásokhoz (03.16)
4. Bejelentkezési hibák javítása (03.18)

* 2025.03.23–03.26

1. Feladatkártyák megnyitása
2. Tanár nézet: diákonként kiosztott feladatok listázása
3. Diák nézet: kiosztott feladatok listázása

* 2025.04.01–04.02

1. Tanár letöltheti a feltöltött fájlokat (04.01)
2. 'Visszaküldve' jelölés megvalósítása (04.02)

* 2025.04.09–04.10

1. Hibák javítása és kinézet finomítása, 2 fázisban

* 2025.04.11

1. Korosztály szerinti szűrés bevezetése
2. Hitelesítés hibáinak javítása (merge: auth-fix branch)

* 2025.04.14

1. Visszaküldött feladatok kezelése javítva (returnedTasks fix)

## 3. Tesztelési Terv

### Hitelesítés

* Tesztnév: Regisztráció tesztelése

Teszt leírása: A rendszer regisztrációs oldala működik és megfelelő adatokat fogad el a különböző szerepkörök (tanár/diák) számára.

Elvárt eredmény: A regisztrációs űrlap sikeresen elmenti a bejegyzett adatokat, és visszaigazolást ad.

* Tesztnév: Bejelentkezési teszt

Teszt leírása: Helyes felhasználónév és jelszó esetén a rendszer belépteti a felhasználót.

Elvárt eredmény: A felhasználó sikeresen bejelentkezik, és a megfelelő főoldalra kerül (diák/ tanár).

* Tesztnév: Helytelen bejelentkezés tesztelése

Teszt leírása: Hibás felhasználónév vagy jelszó esetén a rendszer megfelelő hibaüzenetet ad.

Elvárt eredmény: A felhasználó nem tud belépni, és egyértelmű hibaüzenet jelenik meg.

### Feladatkezelés

* Tesztnév: Feladatok kiosztása tanárként

Teszt leírása: A tanár képes feladatokat kiosztani a diákok számára.

Elvárt eredmény: A rendszer lehetővé teszi a feladatok kiosztását, és a diákok látják azokat a saját felületükön.

* Tesztnév: Feladatok megtekintése és visszaküldése diákként

Teszt leírása: A diákok képesek megtekinteni a számukra kiosztott feladatokat, majd visszaküldeni azokat.

Elvárt eredmény: A diákok visszaküldhetik a feladatokat, a tanár számára látható lesz a „Visszaküldött feladatok” menüpont alatt.

* Tesztnév: Letöltési funkció tesztelése

Teszt leírása: A tanár képes letölteni a feltöltött fájlokat.

Elvárt eredmény: A letöltési funkció működik, és a fájlok letölthetők a megfelelő tanárok számára.

### UI és navigáció

* Tesztnév: Menü aktív állapotának tesztelése

Teszt leírása: A navigációs sáv aktív linkjeinek megfelelő működése.

Elvárt eredmény: Az aktív menüpontok megfelelően kiemeltek, amikor a felhasználó az adott oldalra navigál.

* Tesztnév: Reszponzivitás tesztelése

Teszt leírása: Az alkalmazás minden eszközön és képernyőméreten megfelelően jelenik meg.

Elvárt eredmény: A weboldal reszponzív módon alkalmazkodik a különböző képernyőméretekhez (mobil, tablet, desktop).

* Tesztnév: Hibakezelés tesztelése

Teszt leírása: A hibák kezelése, mint például a nem található oldal (404) vagy hibás API hívás.

Elvárt eredmény: A rendszer megfelelő hibát jelez a felhasználó számára, amikor valami nem működik, és lehetőséget biztosít a visszalépésre vagy újrapróbálkozásra.

### Automatizált tesztek

* Tesztnév: Diák név megjelenítése a navigációs sávban

Teszt leírása: A teszt ellenőrzi, hogy a diák neve megfelelően megjelenik-e a navigációs sávban.

Kód:

test('renders student name from localStorage', () => {

    localStorage.setItem('studentName', 'Test Student');

    render(<StudentPageNav />);

    const nameEl = screen.getByTestId('student-name');

    expect(nameEl).toBeInTheDocument();

    expect(nameEl).toHaveTextContent('Test Student');

  });

Tesztnév: Diák regisztráció és bejelentkezés tesztelése

Teszt leírása: Az automatizált teszt biztosítja, hogy a diákok regisztrálni és bejelentkezni tudjanak.

Kód:

test('student registration and login', async () => {

    render(<RegistrationForm />);

    fireEvent.change(screen.getByLabelText('Username'), { target: { value: 'student1' }});

    fireEvent.change(screen.getByLabelText('Password'), { target: { value: 'password123' }});

    fireEvent.click(screen.getByText('Register'));

    await waitFor(() => screen.getByText('Welcome, student1'));

    expect(screen.getByText('Welcome, student1')).toBeInTheDocument();

  });

Tesztnév: Assignments és Rating funkciók tesztelése

Teszt leírása: A feladatkezelés (Assignments) és tanár értékelés (Rating) komponensek automatizált tesztjei.

Elvárt eredmény:   
- Ellenőrzés, ha nincs tanár:

expect(screen.getByText('Még nem választottál tanárt!'))

- Tanár adatainak megjelenítése:

expect(screen.getByText('Jane Doe'))

- Értékelés leadása után:

expect(screen.getByText('Köszönjük hogy értékeltél!'))

## Tesztelési eredmények

Minden teszt eredménye nyomon követhető, és az automatizált tesztek minden új változtatás után futtatásra kerülnek, hogy biztosítsák a funkciók működőképességét.