Digitális eseménytervező rendszer - Szakdolgozat

Készítette: Béres Ákos Iván
Témavezető: Dr. Bilicki Vilmos
Egyetem: Szegedi Tudományegyetem
Szak: Programtervező informatikus
Év: 2025

Tartalomjegyzék

1. Bevezetés

A mai világban az események tervezése, különösen nagyobb csoportok esetén, egyre összetettebbé válik. A 'Digitális eseménytervező rendszer' egy modern megoldás, amely lehetővé teszi események létrehozását, résztvevők kezelését és feladatok kiosztását.

Az első verzióban a cél az autentikációs modul megvalósítása egy saját backend rendszer segítségével, amely a felhasználók regisztrációját, bejelentkezését és tokenizálását biztosítja. A későbbi verziókban a projekt bővül majd az események hozzáadásával illetve Firebase alapú backend platformra váltással.

2. Architektúra

2.1. Rendszer komponensei

A rendszer két fő komponensből áll:

- Frontend: Angular keretrendszer, amely biztosítja a felhasználói élményt és az interakciókat.

- Backend: Node.js/Express.js alapú REST API, amely az adatok feldolgozásáért és hitelesítéséért felel, illetve MongoDB adatbázis.

2.2. Adatfolyam

1. A felhasználó regisztrációt vagy bejelentkezést kezdeményez.

2. Az Angular frontend továbbítja az adatokat a Node.js backendnek.

3. A backend feldolgozza az adatokat, és JWT tokent generál.

4. A frontend a tokent tárolja (localStorage vagy sessionStorage), és a további API-kérésekhez használja.

3. Funkcionális specifikáció

3.1. Regisztráció

- Bemenet: Felhasználó e-mail cím és jelszó megadása.

- Funkciók:

  - Adatok validálása frontend és backend szinten.

  - Jelszó titkosítása bcrypt segítségével.

  - Adatok mentése az adatbázisba.

- Kimenet: Sikeres regisztráció esetén a rendszer tárolja az adatokat, és értesíti a felhasználót.

3.2. Bejelentkezés

- Bemenet: Felhasználó e-mail címe és jelszava.

- Funkciók:

  - Jelszó ellenőrzése bcrypt-tel.

  - JWT token generálása.

  - Token visszaadása a frontendnek.

- Kimenet: Sikeres bejelentkezés esetén a felhasználó hozzáfér a védett funkciókhoz.

4. Technológiai háttér

4.1. Backend

- Node.js: Futtatási környezet a szerveroldali kódhoz.

- Express.js: Egyszerű keretrendszer REST API-k készítéséhez.

- JWT: Biztonságos token alapú hitelesítés.

- Adatbázis: MongoDB.

4.2. Frontend

- Angular: Komponens alapú frontend fejlesztés.

- SCSS: Testreszabható stílusok reszponzív dizájnhoz.

5. Biztonság

- Jelszó titkosítása: Bcrypt használata.

- Token időkorlát: Érvényességi idő (ebben az esetben 1 óra).

- HTTPS használat: Biztonságos adatkommunikáció.

- Hibakezelés: Biztonságos API válaszok hibás kérésekre.

6. Fejlesztési folyamat

6.1. Iterációk

1. Autentikációs modul fejlesztése.

2. Frontend és backend összekapcsolása.

3. Felhasználói felület fejlesztése.

6.2. Eszközök

- VS Code: Fejlesztési környezet.

- Git: Verziókezelés.

7. Eredmények és továbbfejlesztési lehetőségek

8.1. Eredmények

- Az autentikációs modul sikeresen megvalósítja a felhasználói regisztrációt és bejelentkezést.

8.2. Továbbfejlesztési lehetőségek

- Firebase migráció.

- Események létrehozása és kezelése.

- Értesítési rendszer fejlesztése.

8. Összegzés

Az alkalmazás fejlesztésének első verziója biztosítja a felhasználói hitelesítés biztonságos alapjait. A rendszer később kiegészíthető további funkciókkal, például esemény- és értesítéskezeléssel.