

Fejlesztői kézikönyv (Sandwich app)

Bevezetés:

A sandwich app egy online shop, ahol a felhasználó elkészítheti saját szendvicseit, azokat tárolhatja és megrendelheti.

Telepítés:

Keretrendszerek: Angular, bootstrap, google icons

Telepítendőek:

- Node.js: <https://nodejs.org>
- npm install -g @angular/cli
- npm install -g firebase-tools
- npm install -g bootstrap

Git clone:

- git clone (megadott repository)
- npm install

Environment beállítása:

Hozzunk létre a core mappában egy Environments mappát és azon belül egy environment.ts file-t (core/environments/environment.ts). ide másoljuk be a firebase adatait.

```
export const environment = {  
  production: false, // true-ra állítsd, ha a production környezetben vagy  
  
  firebase: {  
    apiKey: "YOUR_API_KEY",  
    authDomain: "YOUR_AUTH_DOMAIN",  
    projectId: "YOUR_PROJECT_ID",  
    storageBucket: "YOUR_STORAGE_BUCKET",  
    messagingSenderId: "YOUR_MESSAGING_SENDER_ID",  
    appId: "YOUR_APP_ID",  
    measurementId: "YOUR_MEASUREMENT_ID"  
  }  
};
```

Adatok kezelése:

Az adatokat a firebase segítségével tároljuk. Létrehozunk egy realtime database-t és authentication-t. Az authentication-nél email és password bejelentkezést válasszuk ki.

Base service:

A base service segítségével csatlakozunk és módosítjuk a firebase realtime database-t.

Fire auth service:

Az auth service segítségével kommunikálunk a firebase authentication-vel.

Funkciók és komponensek részletezése:

Autentikáció (Auth):

- Login: A felhasználó beléptetése az oldalra. Figyelembe veszi, hogy a felhasználónak verifikálva van-e az email címe.
- Register: A felhasználó regisztrálhat az oldalra és egyben kap egy emailt a verifikálásra.
- Reset-password: A felhasználó megváltoztathatja a jelszavát email-en keresztül.
- User-settings: A felhasználó megváltoztathatja az adatait, amit a realtime database tárol.
- Verify-email: A felhasználó új emailt kérhet az email verifikációhoz.

Fő komponensek (Components):

- Home: A felhasználót egy rövid bevezetés után tovább navigálja a create oldalra, ha be van jelentkezve. Ha nincs akkor a login-ra.
- Sandwich(create): A felhasználó létrehozhat szendvicseket és a létrehozott elemeket kilistázza. A listában külön komponensbe helyezzük az elemeket a könnyebb kezelhetőség érdekében. Az elemeknek két fő funkciója van. Az egyik megjeleníti a szendvics összetevőit, a másik lehetőséget ad a kosárhoz való hozzáadásra és a mennyiség kiválasztására.
- Cart: Itt a kosár tartalmát kezeljük. Kilistázza a kosár tartalmát. Az elemeket itt is kihelyezzük egy külön komponensbe. A listában lévő elemeket kezelhetjük. Hozzáadhatunk mennyiséget egészen 20 darabig és csökkenthetjük a mennyiséget egészen egy-ig. Amikor a darabszám egy és csökkenteni szeretnénk a darabszámot, akkor az elem törlődik a listából. A place order megnyomásával egy újabb komponenst hívunk meg ami a végleges rendeltet mutatja. Itt visszaléphetünk vagy leadhatjuk a rendeltet, ami a realtime database-ben eltárolódik és az orders komponensben kilistázzódik.
- Orders: Itt kilistázzuk a leadott rendelteteket. Az adatokat a realtime database-ről kérjük le. A lekért adatokban figyelembe veszi az elemek státuszát. És

három külön tömbbe helyezi őket. A három tömb alapján listázzuk ki az elemeket (pending, preparation, onTheWay).

Guard:

A guard segítségével blokkoljuk az app-routing-ot a felhasználó bejelentkezése szempontjából.

Navbar:

Egy navigációs komponens, ami a felhasználót segíti navigálni az appon belül. Figyelembe veszi, hogy a felhasználó be van-e jelentkezve. Egyes elemei csak akkor jelennek meg, ha a felhasználó be van jelentkezve.

Jövőbeli fejlesztések:

- Létrehozni egy admin felületet, ahol a felhasználók hozzáférését lehet kezelni.
- Dolgozói felületet létrehozni, ahol a rendeléseket lehet kezelni.