Cukrászda

Java alkalmazások gyakorlat

Balázs Márk | Mérnökinformatikus hallgató

Engler Dániel | Mérnökinformatikus hallgató

2022/2023 1. félév

Tartalomjegyzék

A feladat leírása4

Előkészületek4

A mappa és fájlszerkezet bemutatása név és funkció szerint5

Java fájlok és annak szerkezete5

CSS, HTML, valamint a képek szerkezete5

A tesztelési terv és a tesztelés eredményei6

CURL paranccsal végzett tesztek6

Postman programmal végzett tesztek7

Képernyőképek az alkalmazásról9

Főoldal9

Sütik oldat9

Kapcsolat oldal10

Regisztrációs oldal10

Belépés oldal11

Üzenetek oldal11

Potenciális fejlesztési lehetőségek12

Használt külső állományok/keretrendszerek/sablonok12

**A feladat leírása és a megvalósítás**

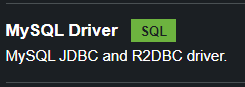
Feladatunk az volt, hogy készítsünk weboldalt egy fiktív cég számára a kapott adatbázis szerkezet segítségével.

* A Feladat publikusan elérhető a Githubon: [**https://github.com/balmrk/Cukraszda**](https://github.com/balmrk/Cukraszda)
  + Balázs Márk munkája: <https://github.com/balmrk/Cukraszda/commits?author=balmrk>
  + Engler Dániel munkája: <https://github.com/balmrk/Cukraszda/commits?author=Engi16>
* A kész Project megtekinthető legkésőbb 2023.januárig a következő linken: [**https://cukraszda.azurewebsites.net/**](https://cukraszda.azurewebsites.net/)

**Előkészületek**

Első lépésként felosztottuk az elvégzendő feladatokat. A kiválasztott adatbázist leíró táblákat txt formátumban kaptuk, ezt először CSV fájllá konvertáltam, majd phpmyadmin segítségével egy MySQL szerver alapjául szolgált. A kapott SQL fájlt feltöltöttem Githubra. (az adatbázishoz tartozó txt, csv, sql fájlok elérhetők a következő linken: <https://github.com/balmrk/Cukraszda/tree/main/Cukraszda_DB> ).

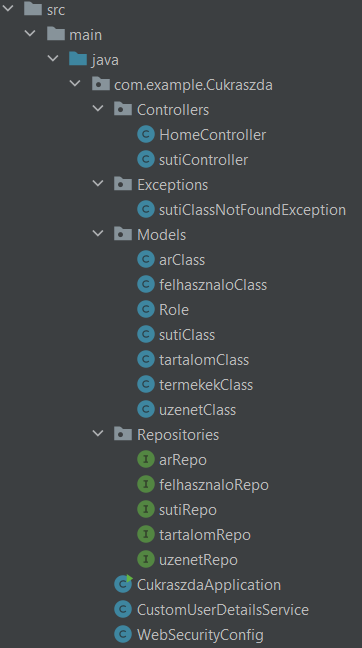
A projecthez a következő függőségeket állapítottam meg:

* 
* A fejlesztéshez MySQL adatbázist használtam, ezért szükséges volt még egy függőség:
* 
* Ezt a végleges verzióból eltávolítottam.

**Megvalósítás**

|  |  |
| --- | --- |
| **Feladatrész** | **Megvalósítás az alkalmazásban** |
| Látványos weboldal | Az index oldalon rövid leírás, tiszta HTML kóddal. A sütik oldalon táblázatos megjelenítés adatbázisból Thymeleaf alkalmazásával. |
| Regisztráció, Bejelentkezési lehetőség | Látogató jogosultságú felhasználó regisztrálhat és beléphet. Az új regisztrációval User rangú felhasználó jön létre (nem jár extra jogosultsággal, de az elküldött üzenetnél látszódik a neve, mint küldő). |
| felhasználói szerepek | Az üzenetek oldal csak az admin jogú felhasználóknak elérhető. |
| Űrlap, Szerver oldali validáció | Regisztrációs oldalon, Bejelentkezési oldalon és Üzenet küldése oldalon. |
| Felhasználó nevének lekérdezése | Menü sávban. |
| Adatbázis írás/olvasás | Írás: Üzenet küldésekor, regisztráláskor.  Olvasás: Sütik oldal betöltése, bejelentkezés, admin oldalon üzenetek megjelenítése. |
| GitHub használat | Main branch-et használva, minden commit esetén rövid leírással a változásokról. Az issues oldalon a feladatokat listáztuk, elvégzésük után lezárttá tettük (ha szükséges volt egy kommenttel együtt). |
| Megvalósítás Interneten a Microsoft Azure felhőben | A MySQL adatbázis átkonvertálása, a Dependency cserélése és a Projectben használt Java verzió módosítása után (17->11) Github klónozással feltöltöttem a Hallgatói profilomba a Java projectet, majd létrehoztam az azure felhőben elérhető Postgre adatbázist. Ehhez a saját gépemen elérhető pgadmin szoftverrel kapcsolódtam és a módosított DDL és DML parancsokat lefuttattam. Ezután Üzembe helyeztem az Azure felhőn az alkalmazást. |
| RESTful API | A sütik tábla menedzselése. |

**A mappa és fájlszerkezet bemutatása név és funkció szerint**

**Mappa- és fájlszerkezet**

Az átláthatóság érdekében a java állományokat kategóriákban rendszereztük.

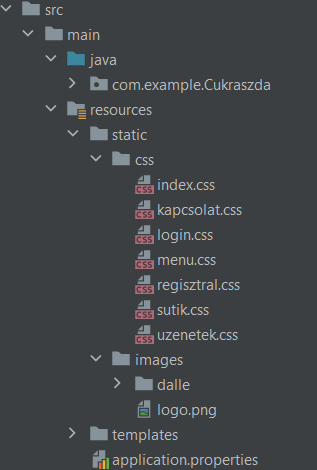
A main metódust tartalmazó CukraszdaApplication közvetelenül a **com.example.Cukraszda** csomagban található, mellette a SpringSecurity-hez szükséges **WebSecurityConfig** (authorizációhoz szükséges konfigurációkat tartalmazza) és **CustomUserDetailsService** (a felhasználói engedélyek ellenőrzésére használt).

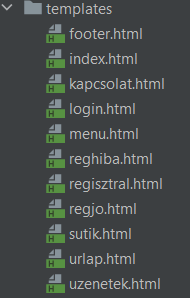
A **Controllers** mappában a vezérlők találhatók. A **HomeController** az alkalmazáson belüli navigációkért felel, a **sutiController** pedig a RESTful működés megvalósítását teszi lehetővé.

Az **Exceptions** mappában egy kivétel osztály szerepel (**sutiClassNotFoundException**), amely egy nem létező süti lekérésekor kerül futásra.

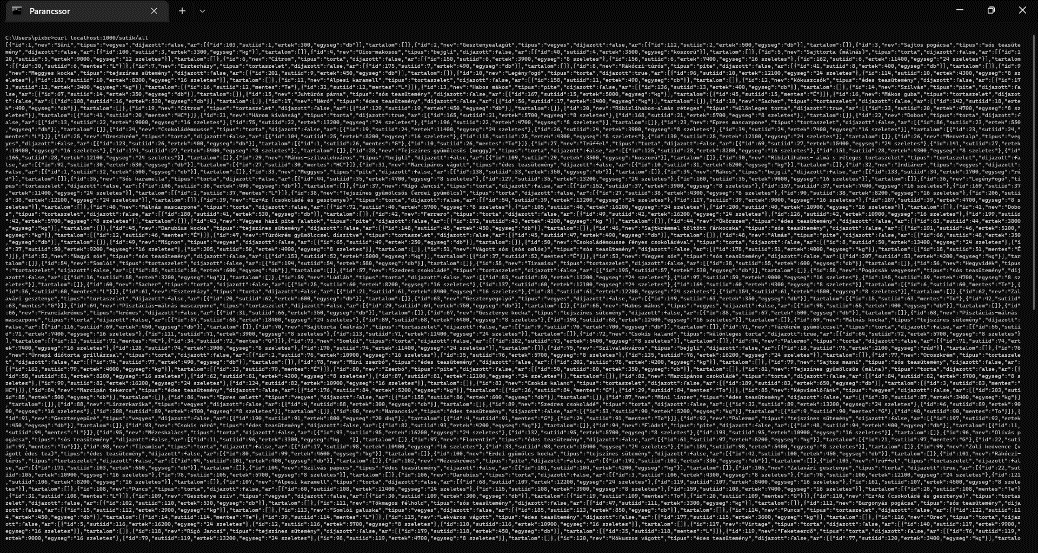
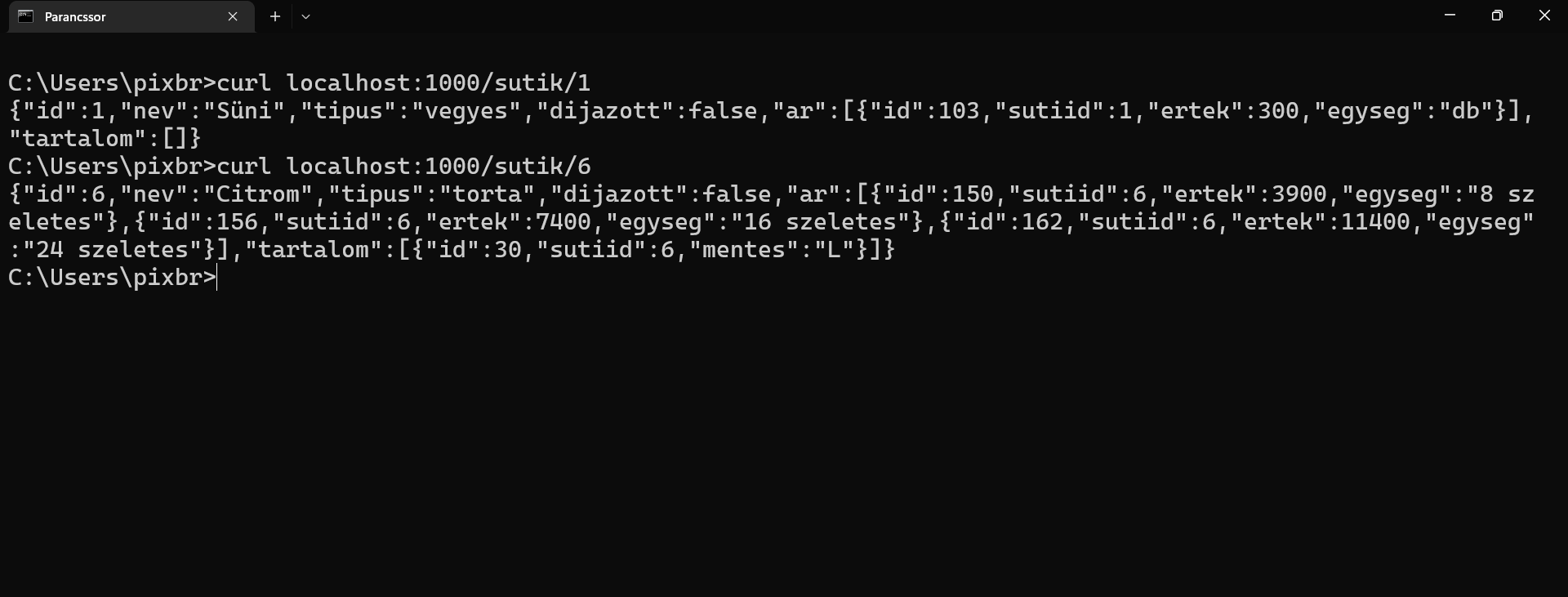
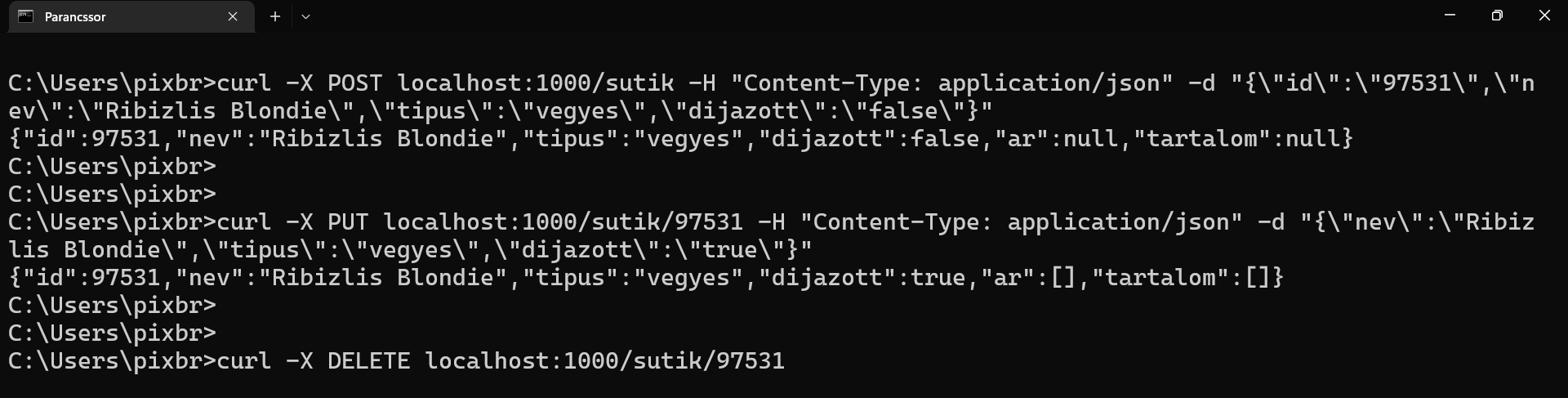
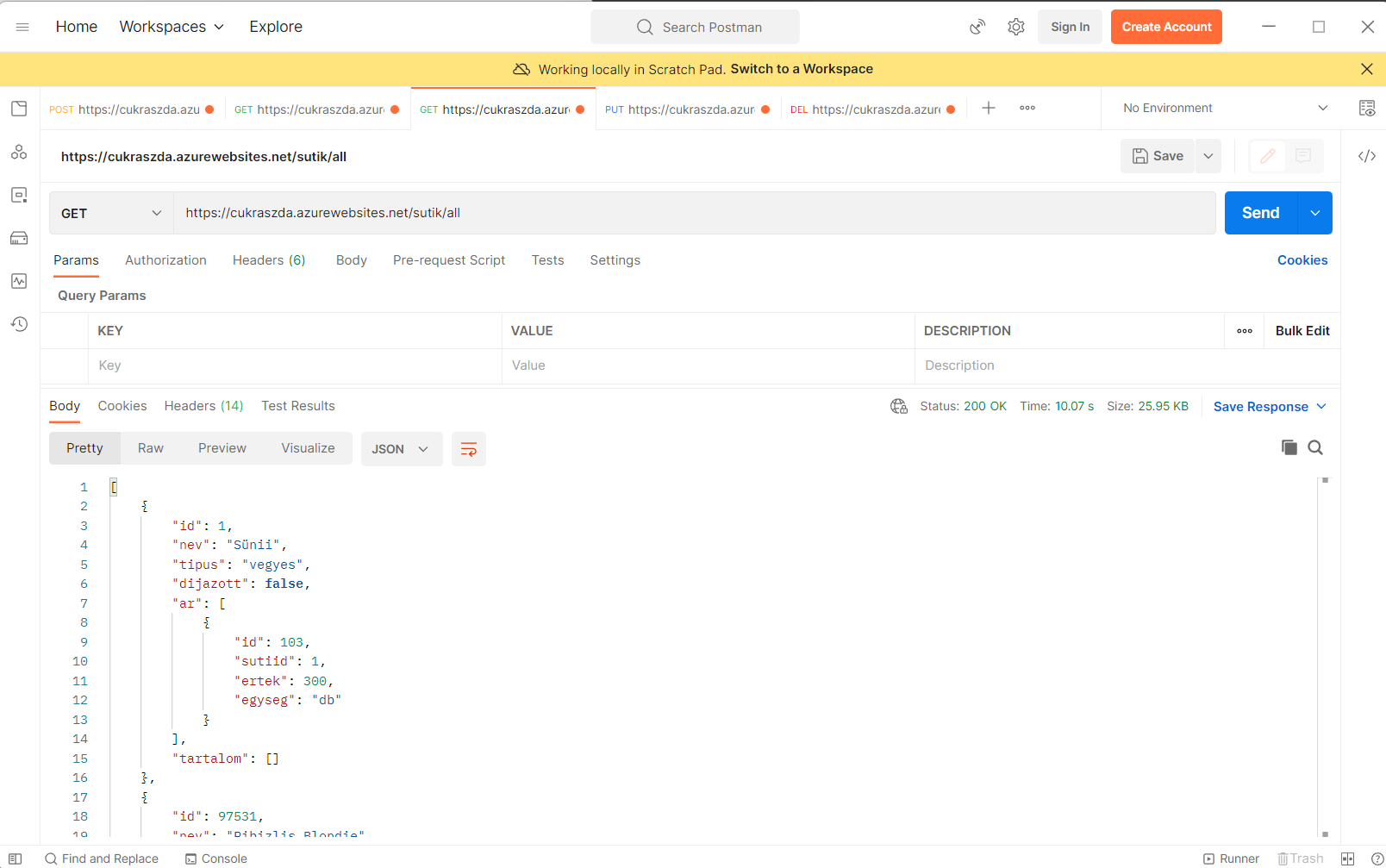
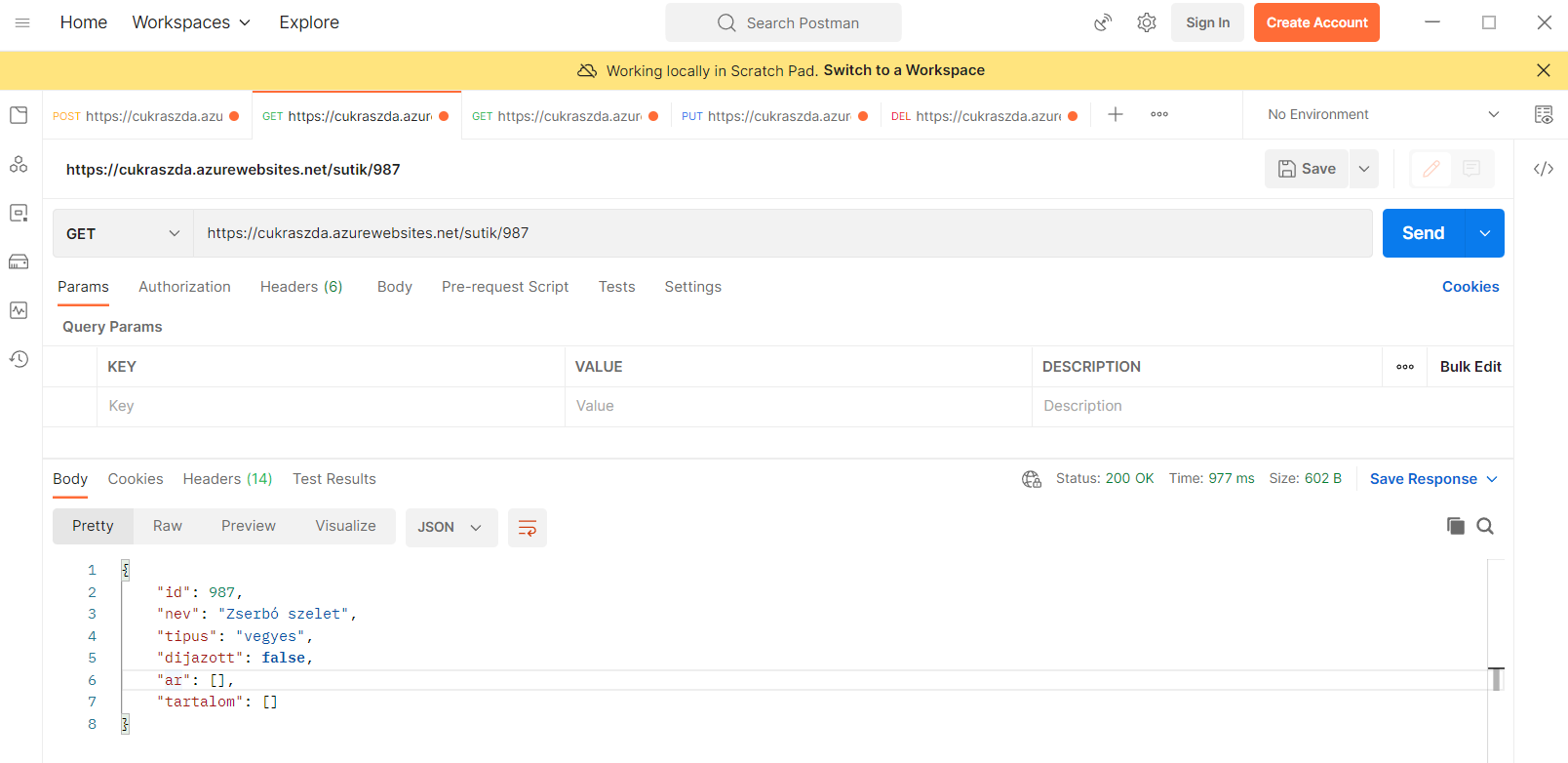
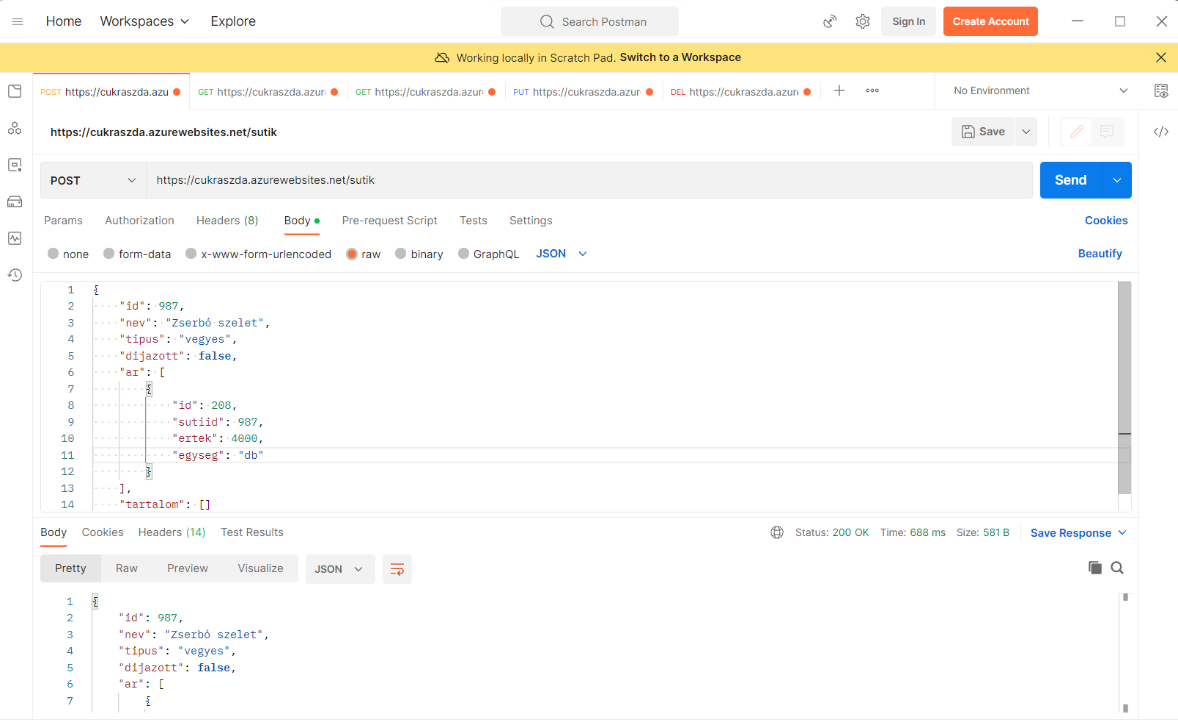
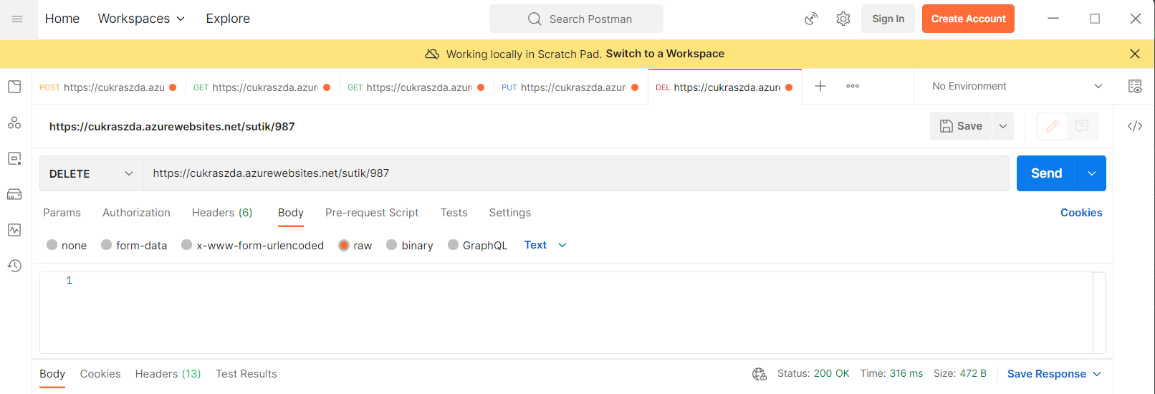
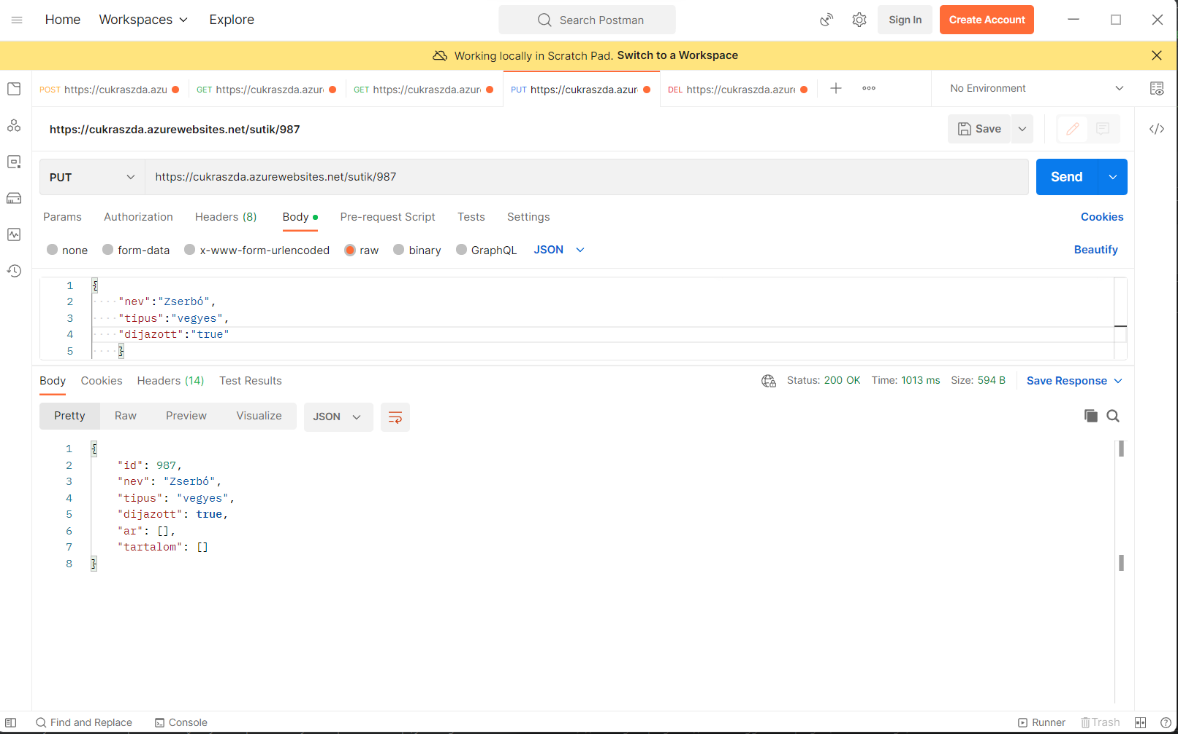
A **Models** mappában az adatbázis kapcsolathoz szükséges Entitás osztályok szerepelnek.

A **Repositories** mappa tartalma szintén az adatbázis kapcsolathoz szükséges Interface állományokat tartalmazza.

A **resources** mappában találhatók a statikus(**static**) állományok: Ezek CSS fájlok és a HTML oldalakon használt képek. A HTML állományok a **templates** mappában találhatók.



**A tesztelési terv és a tesztelés eredményei**

* **CURL tesztek (localhost):**
  + GET ALL****
  + GET BY ID
  + POST, PUT és DELETE
* **POSTMAN tesztek (azurewebsites)**
  + GET ALL
  + GET BY ID****
  + POST
  + PUT
  + DELETE

**Képernyőképek az alkalmazásról**

**Főoldal**

**A képen szöveg látható

Automatikusan generált leírás**

**Sütik oldal**

****

**Kapcsolat oldal**

****

**Regisztrációs oldal**

****

**Belépés oldal**

****

**Üzenetek oldal (Csak admin jogosultságú felhasználó látja)**

**A képen asztal látható

Automatikusan generált leírás**

**Potenciális fejlesztési lehetőségek**

* Új/más dizájn
* Szerveroldali validáció
* Dinamikusabb web

**Használt külső állományok/keretrendszerek/sablonok**

A gyakorlaton vett módszereket alkalmaztuk kiinduló pontnak. Alkalmanként felmerülő akadályok esetén a Stackoverflow fórum oldalon kerestünk segítséget.