

Jakub Abrahoim, Adam Blahovič

Zadanie 2 – míľnik 1

Dokumentácia

Mobilné technológie a aplikácie

Cvičenie: Streda, 18:00

Cvičiaci: Ing. Matej Janeba

Obsah

Obsah	2
Návrh databázy	3
Wireframy	4
Testy	6
Frontend	6
Kladné	6
Záporné	7
Backend	8
Kladné	8
Záporné	8
REST API	10
Vyhľadať jedlo - GET	10
Vyhľadať potravinu - GET	10
Zobraziť log - GET	11
Zobraziť detaily jedla - GET	11
Zobraziť detaily potraviny - GET	12
Zobraziť používateľove informácie - GET	12
Pridať jedlo - POST	13
Pridať potravinu - POST	13
Pridať jedlo do logu - POST	14
Prihlásiť sa - POST	14
Registrovať sa - POST	15
Upraviť jedlo - PUT	15
Upraviť potravinu - PUT	16
Upraviť položku v logu - PUT	16
Upraviť používateľské informácie - PUT	17
Vymazať jedlo - DELETE	17
Vymazať potravinu - DELETE	18
Vymazať položku v logu - DELETE	18
Vymazať používateľa - DELETE	18
Changelog	19
Endpointy	19
Databáza	19
Implementačný opis	19

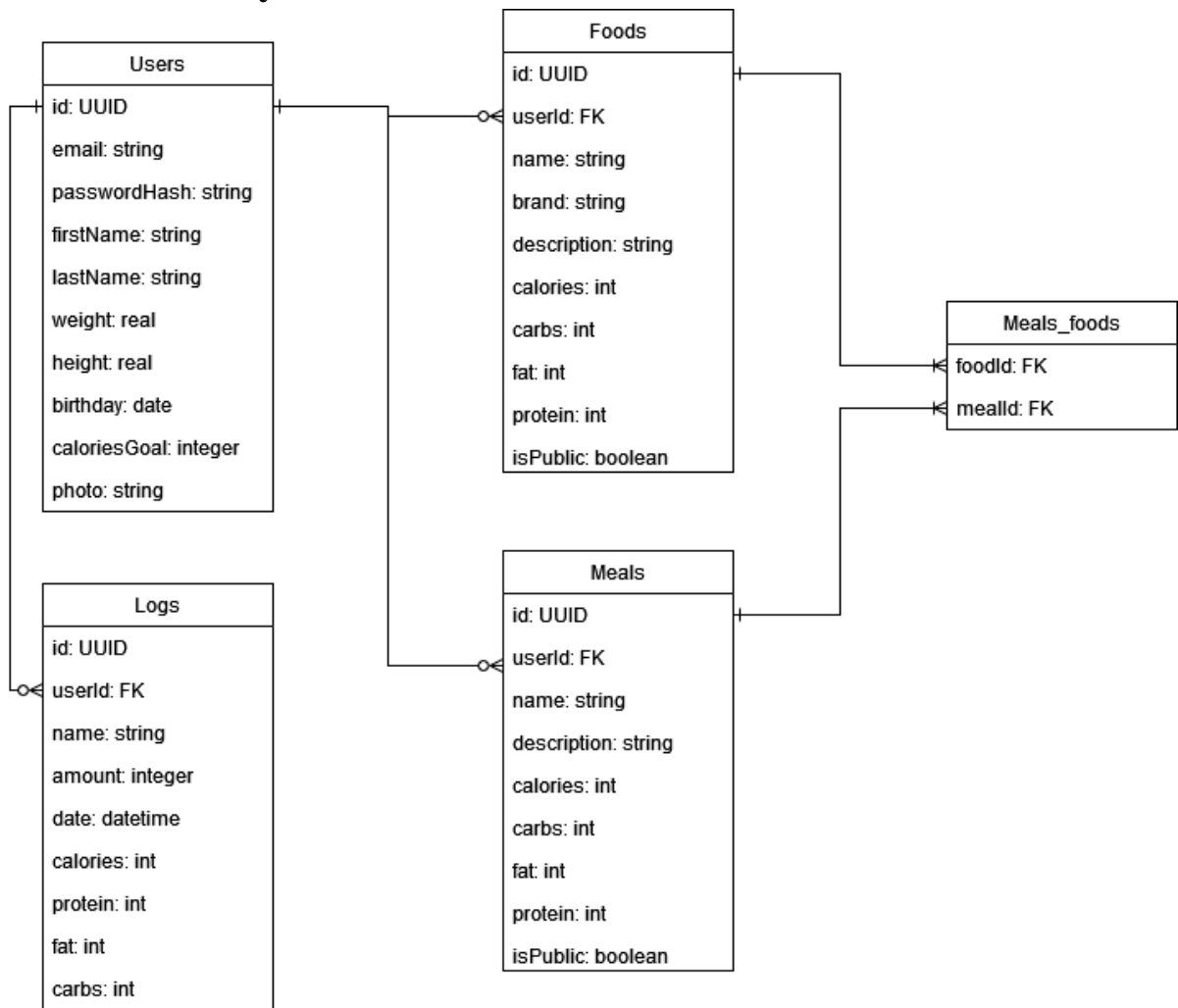
Odkaz na Notion dokument s lepším formátovaním:

<https://jakub-abrahoim.notion.site/Zadanie-2-M-nik-1-Jakub-Abrahoim-Adam-Blahovi-371de87aebf24dae8b4ac661b72713f8>

Odkaz na repozitár:

<https://github.com/FIIT-ness-pal/backend>

Návrh databázy



Wireframy

FIITness Pal

Email

Heslo

Heslo znovu

Meno

Priezvisko

Dátum narodenia

Výška cm Váha kg

Denný kalorický cieľ cal

Registovať sa

FIITness Pal

Email

Heslo

Prihlásiť sa

Nemáte účet?

Registovať sa

Dobrý večer, Ivan!

Čo ste mali na večeru?

Pridať

Váš dnešný príjem

360 / 2000

90 40 90

Sacharidy Tuky Proteíny

História

6. 3. 2022

8:00 Jogurt 50 cal

12:00 Špagety 300 cal

☎ 🏠 + 🔍 👤

< Späť

Profil

Fotka:

Meno: Nikola Kotes

Výška: 150 cm

Váha: 150 kg

Kalórie: 2000 cal

E-mail: ivko69@pokec.sk

Heslo: ***

Uložiť

☎ 🏠 + 🔍 👤

< Späť

Vyhľadávanie

Jedlá	Potraviny
Špagety 300 cal	Zobraziť
Kuracie prsia s ryžou 300 cal	Zobraziť

☎ 🏠 + 🔍 👤

< Späť

Uložiť

Názov jedla

Ivkove špagety

Popis

Najlepšie špagetky

Hodnoty

360 90 40 90

Kalórie Sacharidy Tuky Proteíny

Ingrediencie

Špagety Podravka 300 cal	Množstvo	×
Morca-della Hamé 300 cal	Množstvo	×

Pridať potravinu

Zverejniť pre všetkých ☐

☎ 🏠 + 🔍 👤

< Spät'

Pridať

Ivkove špagety

od Ivan Bežec

360904090

KalórieSacharidyTukyProteíny

Popis:

Najlepšie špagety aké som kedy jedol

Ingrediencie

Špagety

Podravka300 cal

50g

Morca-della

Hamé300 cal

100g

+

< Spät'

Uložiť

Názov potraviny

Špagety

Značka

Barilla

Hodnoty

Kalórie

Množstvo

Sacharidy

Množstvo

Tuky

Množstvo

Proteíny

Množstvo

Zverejniť pre všetkých☐

+

< Spät'

Pridať

Tvaroh ochutený Jablko-škorica

Rajo

360904090

KalórieSacharidyTukyProteíny

Množstvo

g

+

Testy

Frontend

Kladné

1. Úspešné prihlásenie

Vstupné podmienky:	Používateľ má otvorenú aplikáciu na stránke s prihlásením a je pripojený na Internet
Výstupné podmienky:	Používateľ sa presunie na domovskú stránku aplikácie
Postup:	1. Používateľ vyplní pole pre e-mailovú adresu 2. Používateľ vyplní pole pre heslo 3. Používateľ klikne na tlačidlo Prihlásiť sa 4. Používateľ je presmerovaný na domovskú stránku
Výsledok:	PASS

2. Vytvorenie jedla

Vstupné podmienky:	Používateľ je prihlásený a nachádza sa na domovskej stránke. Aplikácia je pripojená na Internet.
Výstupné podmienky:	V aplikácii pribudne nové jedlo, ktoré sa uloží do databázy.
Postup:	1. Používateľ stlačí tlačidlo Pridať a zvolí možnosť Vytvoriť jedlo 2. Zobrazí sa obrazovka na vytvorenie jedla, kde používateľ vyplní názov jedla a pridá popis 3. Používateľ klikne na tlačidlo Pridať potravinu 4. Zobrazí sa obrazovka s vyhľadávaním potravín 5. Používateľ vyhľadá požadovanú potravinu, klikne na tlačidlo Zobrazit', ktorá ho presmeruje na obrazovku detailu potraviny 6. Používateľ klikne na tlačidlo pridať, kde zadá množstvo a výber potvrdí 6. Opäť sa zobrazí obrazovka na vytvorenie jedla 7. Používateľ opakuje kroky 3. až 6., kým nepridá všetky potraviny. 8. Používateľ klikne na tlačidlo Uložiť
Výsledok:	PASS

3. Log jedla

Vstupné podmienky:	Používateľ je prihlásený a nachádza sa na domovskej stránke. Aplikácia je pripojená na Internet.
Výstupné podmienky:	Do používateľovho logu pribudne nové jedlo a prepočíta sa príjem jeho kalórií za daný deň
Postup:	1. Používateľ stlačí tlačidlo Pridať a zvolí možnosť Pridať jedlo do logu 2. Zobrazí sa obrazovka s vyhľadávaním 3. Používateľ napíše do vyhľadávača názov jedla 4. Používateľ vyhľadá požadovanú potravinu, klikne na tlačidlo Zobrazit', ktorá ho presmeruje na obrazovku detailu potraviny 5. Používateľ klikne na tlačidlo pridať, kde zadá množstvo a výber potvrdí 6. Zobrazí sa domovská stránka
Výsledok:	PASS

Záporné

1. Neúspešná registrácia

Vstupné podmienky:	Používateľ má otvorenú aplikáciu na stránke s prihlásením a je pripojený na Internet
Výstupné podmienky:	Používateľovi sa nepodarí zaregistrovať a zostane na obrazovke s registráciou
Postup:	1. Používateľ klikne na tlačidlo registrovať sa a zobrazí sa obrazovka s registráciou 2. Používateľ vyplní všetky potrebné polia 3. Do poľa pre kontrolu hesla zadá nesprávne heslo 4. Používateľ klikne na tlačidlo Registrovať sa 5. Zobrazí sa červený text "Heslá sa nezhodujú"
Výsledok:	FAIL

2. Neúspešné vyhľadávanie

Vstupné podmienky:	Používateľ má otvorenú aplikáciu na stránke s vytváraním jedla a je pripojený na internet
Výstupné podmienky:	Používateľovi sa nepodarí vyhľadať potravinu a zostane na obrazovke s vyhľadávaním
Postup:	1. Používateľ klikne na tlačidlo Pridať potravinu 2. Zobrazí sa obrazovka vyhľadávania 3. Používateľ do poľa na vyhľadávanie zadá potravinu, ktorá sa nenachádza v databáze 4. Na mieste, kde sa bežne zobrazujú výsledky vyhľadávania, sa vypíše "Nenašla sa žiadna zhoda"
Výsledok:	FAIL

Backend

Kladné

1. Prihlásenie do aplikácie

Vstupné podmienky:	Server je spustený a pripojený na databázu
Výstupné podmienky:	Server verifikoval používateľa a vrátil jeho dáta
Postup:	1. Na server príde požiadavka POST /login s prihlasovacími údajmi 2. Server autentifikuje používateľa 3. Server vráti status kód 200 4. Na server príde požiadavka GET /log s dnešným dátumom a ID používateľa 5. Server odpovie s danými údajmi
Výsledok:	PASS

2. Pridanie jedla do logu

Vstupné podmienky:	Server je spustený a pripojený na databázu
Výstupné podmienky:	Do databázy sa pridá záznam s jedlom
Postup:	1. Na server príde požiadavka GET /meal 2. Server odpovie kódom 200 a vráti údaje o danom jedle 3. Na server príde požiadavka POST /log s údajmi o jedle a ID používateľa 4. Do databázy sa uloží nový záznam 5. Server vráti status kód 200
Výsledok:	PASS

3. Upravenie používateľovej hmotnosti

Vstupné podmienky:	Server je spustený a pripojený na databázu
Výstupné podmienky:	V databáze sa zmení záznam s váhou
Postup:	1. Na server príde požiadavka GET /user na používateľove údaje 2. Server odpovie s danými údajmi 3. Na server príde požiadavka PUT /user s upravenými údajmi 4. Do databázy sa vloží upravená hodnota 5. Server odpovie kódom 200
Výsledok:	PASS

Záporné

1. Neúspešná registrácia - email už existuje

Vstupné podmienky:	Server je spustený a pripojený na databázu
Výstupné podmienky:	Server vráti chybový status kód
Postup:	1. Na server príde požiadavka POST /register s registračnými údajmi 2. Server skontroluje, či sa v databáze nachádza používateľov e-mail 3. Server vráti kód 409 (Conflict)
Výsledok:	FAIL

2. Neúspešné prihlásenie - nesprávne heslo

Vstupné podmienky:	Server je spustený a pripojený na databázu
Výstupné podmienky:	Server vráti chybový status kód
Postup:	1. Na server príde požiadavka POST /login s prihlasovacími údajmi 2. Server skontroluje, či prijaté údaje súhlasia s údajmi v databáze 3. Server vráti kód 401 (Unauthorized)
Výsledok:	FAIL

3. Prázdné pole pri úprave jedla

Vstupné podmienky:	Server je spustený a pripojený na databázu
Výstupné podmienky:	Server vráti chybový status kód
Postup:	1. Na server príde požiadavka GET /meal 2. Server vráti údaje o danom jedle a kód 200 (OK) 3. Na server príde požiadavka PUT /meal s ID jedla a dátami, ktoré sa majú zmeniť. Jedno z polí je prázdne 4. Server vráti kód 400 (Bad Request) a údaje v databáze sa nezmenia
Výsledok:	FAIL

4. Nesprávny formát pri úprave používateľských údajov

Vstupné podmienky:	Server je spustený a pripojený na databázu
Výstupné podmienky:	Server vráti chybový status kód
Postup:	1. Na server príde požiadavka GET /user s ID používateľa 2. Server vráti údaje o používateľovi a kód 200 (OK) 3. Na server príde požiadavka PUT /user s novými údajmi o používateľovi, parameter váha nie je číslo 4. Server vráti kód 400 (Bad Request) a údaje v databáze sa nezmenia
Výsledok:	FAIL

REST API

Vyhľadaj jedlo - GET

Request

```
[GET endpoint] /meals  
[Príklad] /meals?name=spagety
```

Parameters

- **name** (string) - text, ktorý bol zadáný do vyhľadávacieho poľa

Response

```
{  
  "meals": [  
    {  
      "id": 1,  
      "name": "spagety",  
      "calories": 486,  
      "carbs": 80,  
      "protein": 20,  
      "fat": 10  
    },  
    {  
      "id": 44,  
      "name": "spagety alfredo",  
      "calories": 486,  
      "carbs": 80,  
      "protein": 20,  
      "fat": 10  
    }  
  ]  
}
```

Vyhľadaj potravinu - GET

Request

```
[GET endpoint] /foods  
[Príklad] /foods?name=flakes
```

Parameters

- **name** (string) - text, ktorý bol zadáný do vyhľadávacieho poľa

Response

```
{  
  "foods": [  
    {  
      "id": 1,  
      "name": "Frosted flakes",  
      "brand": "Kellogs",  
      "calories": 486,  
      "carbohydrates": 80,  
      "protein": 20,  
      "fat": 10  
    },  
    {  
      "id": 44,  
      "name": "Corn flakes",  
      "brand": "Kellogs",  
      "calories": 486,  
      "carbohydrates": 80,  
      "protein": 20,  
      "fat": 10  
    }  
  ]  
}
```

Zobrazit' log - GET

Request

```
[GET endpoint] /log
[Příklad] /log?userId=1&date=2022-03-10
```

Parameters

- **userId** (integer) - ID uživatele
- **date** (timestamp) - datum

Response

```
{
  "totalCalories": 650,
  "totalProtein": 50,
  "totalFat": 30,
  "totalCarbs": 50,
  "logs": [
    {
      "name": "jahodový jogurt",
      "amount": 150,
      "time": "7:30",
      "calories": 150
    },
    {
      "mealName": "špagety",
      "amount": 300,
      "time": "12:00",
      "calories": 300
    },
    {
      "mealName": "šalát",
      "amount": 200,
      "time": "16:30",
      "calories": 200
    }
  ]
}
```

Zobrazit' detaily jídla - GET

Request

```
[GET endpoint] /meal
[Příklad] /meal?id=31
```

Parameters

- **id** (integer) - ID jídla

Response

```
{
  "id": 31,
  "name": "špagety",
  "calories": 486,
  "carbs": 80,
  "protein": 20,
  "fat": 10,
  "description": "Moje skvelé špagety",
  "amount": 300,
  "ingredients": [
    {
      "id": 1,
      "name": "cestoviny",
      "brand": "K-classic",
      "amount": 20
    },
    {
      "id": 2,
      "name": "morca-della",
      "brand": "Hame",
      "amount": 20
    }
  ]
}
```

Zobrazit' detaily potraviny - GET

Request

```
[GET endpoint] /food  
[Příklad] /food?id=2
```

Parameters

- **id** (integer) - ID potraviny

Response

```
{  
  "id": 2,  
  "name": "morca-della",  
  "brand": "Hame",  
  "amount": 20,  
  "calories": 486,  
  "carbs": 80,  
  "protein": 20,  
  "fat": 10,  
}
```

Zobrazit' používateľove informácie - GET

Request

```
[GET endpoint] /user  
[Příklad] /user?id=3
```

Parameters

- **id** (integer) - ID používateľa

Response

```
{  
  "firstName": "Peter",  
  "lastName": "Zdravý",  
  "email": "peter.zdravy@pokec.sk",  
  "weight": 100,  
  "height": 185,  
  "birthday": 1995-10-5,  
  "caloriesGoal": 2200,  
  "photo": "https://petkovafotka.com/petko"  
}
```

Pridať jedlo - POST

Request

```
[POST endpoint] /meal
```

Request body

```
{
  "name": "Ivkove špagety",
  "description": "Moje mňam mňam špagetky",
  "userId": 3,
  "amount": 20,
  "calories": 486,
  "carbs": 80,
  "protein": 20,
  "fat": 10,
  "isPublic": true,
  "ingredients": [
    {
      "id": 4,
      "amount": 20,
    },
    {
      "id": 5,
      "amount": 10,
    }
  ]
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 422,
  "message": "Unprocessable Entity"
}
```

Pridať potravinu - POST

Request

```
[POST endpoint] /food
```

Request body

```
{
  "name": "Jablko",
  "description": "Jablko červené",
  "userId": 3,
  "brand": "Veselé jablko zo Žitného ostrova",
  "amount": 5,
  "calories": 486,
  "carbs": 80,
  "protein": 20,
  "fat": 10,
  "isPublic": true,
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 422,
  "message": "Unprocessable Entity"
}
```

Pridať jedlo do logu - POST

Request

```
[POST endpoint] /log
```

Request body

```
{
  "userId": 3,
  "name": "Pizza Hawaii",
  "date": "2022-03-10 15:30",
  "amount": 400,
  "calories": 800,
  "carbs": 80,
  "protein": 20,
  "fat": 300
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 422,
  "message": "Unprocessable Entity"
}
```

Prihlásiť sa - POST

Request

```
[POST endpoint] /login
```

Request body

```
{
  "email": "marek.parek@oharek.com",
  "password": "123styri"
}
```

Responses

```
{
  "status": 200,
  "message": "OK"
}

{
  "status": 401,
  "message": "Unauthorized"
}
```

Registrovať sa - POST

Request

```
[POST endpoint] /register
```

Request body

```
{
  "email": "marek.parek@oharek.com",
  "password": "123styri",
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 409,
  "message": "Conflict"
}
```

Upraviť jedlo - PUT

Request

```
[PUT endpoint] /meal
```

Request body

```
{
  "id": 7,
  "name": "Ivkove špagety",
  "description": "Moje mňam mňam špagetky",
  "userId": 3,
  "amount": 20,
  "calories": 486,
  "carbs": 80,
  "protein": 20,
  "fat": 10,
  "isPublic": true,
  "ingredients": [
    {
      "id": 4,
      "amount": 20,
    },
    {
      "id": 5,
      "amount": 10,
    }
  ]
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 422,
  "message": "Unprocessable Entity"
}
```

Upraviť potravinu - PUT

Request

```
[PUT endpoint] /food
```

Request body

```
{
  "id": 5,
  "name": "Jablko",
  "description": "Jablko červené",
  "userId": 3,
  "brand": "Veselé jablko zo Žitného ostrova",
  "amount": 5,
  "calories": 486,
  "carbs": 80,
  "protein": 20,
  "fat": 10,
  "isPublic": true,
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 422,
  "message": "Unprocessable Entity"
}
```

Upraviť položku v logu - PUT

Request

```
[PUT endpoint] /log
```

Request body

```
{
  "id": 3,
  "name": "Pizza Hawaii",
  "date": "2022-03-10 14:25",
  "amount": 20,
  "calories": 486,
  "carbs": 80,
  "protein": 20,
  "fat": 10
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 422,
  "message": "Unprocessable Entity"
}
```


Upraviť používateľské informácie - PUT Request

```
[PUT endpoint] /user
```

Request body

```
{
  "id": 10,
  "firstName": "Peter",
  "lastName": "Zdravý",
  "email": "peter.zdravy@pokec.sk",
  "weight": 100,
  "height": 185,
  "birthday": 1995-10-5,
  "caloriesGoal": 2200,
  "photoUrl": "https://petkovafotka.com/petko"
}
```

Responses

```
{
  "status": 201,
  "message": "Created"
}

{
  "status": 422,
  "message": "Unprocessable Entity"
}
```

Vymazať jedlo - DELETE Request

```
[DELETE endpoint] /meal
```

Parameters

- **id** (integer) - ID jedla

Responses

```
{
  "status": 200,
  "message": "OK"
}

{
  "status": 404,
  "message": "Not Found"
}
```

Vymazať potravinu - DELETE

Request

```
[DELETE endpoint] /food
```

Parameters

- **id** (integer) - ID potraviny

Responses

```
{
  "status": 200,
  "message": "OK"
}

{
  "status": 404,
  "message": "Not Found"
}
```

Vymazať položku v logu - DELETE

Request

```
[DELETE endpoint] /log
```

Parameters

- **id** (integer) - ID položky

Responses

```
{
  "status": 200,
  "message": "OK"
}

{
  "status": 404,
  "message": "Not Found"
}
```

Vymazať používateľa - DELETE

Request

```
[DELETE endpoint] /user
```

Parameters

- **id** (integer) - ID používateľa

Responses

```
{
  "status": 200,
  "message": "OK"
}

{
  "status": 404,
  "message": "Not Found"
}
```

Changelog

Endpoints

- pridanie endpoint GET,POST /userPhoto na pridanie a zobrazenie profilovej fotky
- /food POST, /meal POST, /log POST, /meal PUT - userId už neposielame, pretože ho vytiahujeme z JSON Web Tokenu
- /food GET, /food POST - amount neposielame, všetko sa prerátava na 100 gramov
- /meal GET, /meal POST, /meal PUT - amount neposielame, pretože je v jednotlivých ingredienciách
- /log GET - k jednotlivým položkám vraciam aj makrá (P/F/C) a nevraciam celkové hodnoty (tie sa prípadne prerátajú na front-ende)
- /register POST - posielame aj používateľove osobné údaje, aby sme rovno vytvorili účet

Databáza

- Tabuľku Meals_foods sme zmenili na Ingredients so stĺpcami amount, mealId a makrami (P/F/C), aby sme neukladali priamy odkaz na Food ale jeho kópiu (v prípade, že sa zmení)
- V tabuľke Log sme stĺpec date rozdelili na date a time, aby sa time dal lepšie spracovať

Implementačný opis

Back-end je implementovaný v jazyku TypeScript s použitím runtime prostredia Node.js a frameworku Express.js. Pomocou Express realizujeme jednotlivé endpointy. Pracujeme s databázou PostgreSQL, kde dopyty generujeme pomocou frameworku TypeORM.

Autentifikáciu používateľov zabezpečuje framework JWT, ktorý generuje a verifikuje web tokeny.

V súbore index.ts sa nachádzajú všetky endpointy, pomocné funkcie sú uložené vo functions.ts. V priečinku entities sú uložené jednotlivé tabuľky databázy, ktoré sa dajú naplniť seedmi v priečinku seeds. Do priečinku images ukladáme používateľské fotky.