

Slovenská technická univerzita v Bratislave  
Fakulta informatiky a informačných technológií

Matej Dudák, Dominik Šalgovič

# Zdieľaný nákupný zoznam

Mobilné technológie a aplikácie

# Špecifikácia

V priebehu semestra ako dvojica vytvoríme aplikáciu pre android podľa zadania. Naša aplikácia bude slúžiť ako zdieľaný to-do list, nákupný zoznam, podľa ktorého by ľudia mohli nakúpiť tým, ktorí sami nemôžu ísť na nákup z dôvodu karantény.

Pre vytvorenie aplikácie sme si zvolili technológiu s programovacím jazykom Kotlin. Backend budeme riešiť pomocou frameworku Django v pythone. Spomenutý framework bude pracovať s PostgreSQL databázou, obsahujúcou tri, navzájom poprepájané tabuľky.

## Názvoslovie

V aplikácií ako aj v dokumentácií používame nasledovné výrazy a pomenovania:

- *Volunteer* - dobrovoľník, ktorý robí nákupy pre druhých
- *Client* - človek, ktorý si sám nemôže nakúpiť, preto vytvára požiadavky (*demands*) pre dobrovoľníkov (*volunteers*)
- *Demand* - požiadavka na nákup vytvorená clientom
- *Demand list* - zoznam požiadaviek, ktoré boli vytvorené používateľmi
- *Expiration date* - dátum, do kedy je daná požiadavka (*demand*) aktuálna

## Navigácia

Aplikácia pozostáva z mnohých obrazoviek. Pomocou aplikácie Framer sme vytvorili interaktívnu vizualizáciu, dostupnú na url: <https://bit.ly/3sWHMkO>

Aplikácia pozostáva z nasledujúcich obrazoviek:

- Login/Signup screen
- Obrazovka s mojím *Demand list*
- Obrazovka s *Demand list* pri ktorých pomáham
- Obrazovka pre vyhľadávanie medzi *Demands* podľa adresy
- Obrazovka pre vytvorenie nového *Demand*
- Obrazovka pre zobrazenie konkrétneho *Demand*
- Vysúvateľná obrazovka obsahujúca informácie o používateľovi

# Databáza

Nami navrhovaná databáza obsahuje nasledujúce tri tabuľky:

## USERS

- **id**
- **user\_name** - unique
- **password**
- **avatar**
- **address**

## DEMANDS

- **demand\_id** - unique, auto increment
- **demand\_title**
- **address**
- **created\_at**
- **expire'd\_at**
- **client\_id** - foreign key
- **volunteer\_id** - foreign key
- **state** - [created, accepted, completed, approved, expired, ...]

## ITEMS

- **item\_id** - unique, auto increment, delete cascade
- **item\_name**
- **quantity**
- **unit** - [pcs, kg, ...]
- **price**
- **demand\_id** - foreign key

# REST Endpointy - Návrh

Naše backend endpointy sú rozdelené naprieč tromi controllermi.

## User controller

### POST /signup

```
{
  "User_name": "test",
  "Password": "password",
  "Address": "address"
}
```

Adresa sa zisti z GPS.

### POST /auth/login

```
{
  "User_name": "test",
  "Password": "password"
}
```

### POST /auth/logout

```
{
  "User_name": "test",
}
```

### PUT /users/user\_name/filename.jpg

### GET /users/user\_name

```
{
  "User_name": "test",
  "Avatar": "url",
  "Address": "address"
}
```

# Demand Controller

## POST /demand

```
{
  "Address": "address",
  "Title": "title",
  "Expired_in": 2,
  "Client_name": "test",
  "Items": [
    {
      "Id": "id",
      "Item_name": "Banány",
      "Number": 2,
      "Unit": "kg"
    }
  ]
}
```

## PUT /demand/{id}

```
{
  "Volunteer_name": "volunteer"
}
```

## GET /demand?filter=&volunteer\_name=&client\_name=

```
{
  [
    {
      "Id": "id",
      "title": "title",
      "address": "address",
      "created_at": "2021-03-07T19:50:26",
      "expired_at": "2021-03-07T19:51:26",
      "Client_name": "test",
      "volunteer_name": "test",
      "State": "accepted"
    }
  ]
}
```

GET /demand/{id}

```
{
  "Address": "address",
  "title": "title"
  "created_at": "2021-03-07T19:50:26",
  "expired_at": "2021-03-07T19:51:26",
  "volunteer_name": "test",
  "Client_name": "test",
  "State": "accepted",
  "items":
  [
    {
      "id": id,
      "item_name": "banany",
      "number": 2,
      "Unit": "kg",
      "price": 5
    }, ...
  ]
}
```

DELETE /demands/{id}

## Item Controller

PUT /item/{id}

```
{
  "Item_name": "Banány",
  "Number": 2,
  "Unit": "kg",
  "Price": 5
}
```

POST /item/{demand\_id}/{item\_id}

```
{
  "Item_name": "Banány",
  "Number": 2,
  "Unit": "kg",
  "Price": 5
}
```

DELETE /item/{id}

# Akceptačné testy

## Frontend

<b>Test 1:</b>	Zobrazenie client a volunteer listu
<b>Vstupné podmienky:</b>	Používateľ je prihlásený v aplikácii a má pripojenie na internet.
<b>Výstupné podmienky:</b>	Používateľovi sa zobrazia v sekcii My Demands iba tie demands, ktoré vytvoril a tie, ktoré akceptoval.
<b>Postup:</b>	<ol style="list-style-type: none"><li>1. Používateľ prejde na sekciu My Demands</li><li>2. Zobrazí sa zoznam demands, ktoré boli vytvorené používateľom</li><li>3. Používateľ prejde na sekciu Volunteer</li><li>4. Zobrazí sa zoznam demands, ktoré používateľ akceptoval</li></ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 2:</b>	Zmena profilovej fotografie
<b>Vstupné podmienky:</b>	Používateľ je prihlásený v aplikácii a má pripojenie na internet.
<b>Výstupné podmienky:</b>	Používateľ bude mať zmenenú profilovú fotografiu.
<b>Postup:</b>	<ol style="list-style-type: none"><li>1. Používateľ klikne na tlačidlo MENU, ktoré sa vyroluje zľava.</li><li>2. Používateľ stlačí ikonu profilovej fotky.</li><li>3. Zobrazí sa externá aplikácia na vybratie fotografie z galérie.</li><li>4. Používateľ vyberie a potvrdí fotografiu.</li><li>5. Zobrazí sa informácia o pridaní fotografie.</li></ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 3:</b>	Nová registrácia existujúceho používateľa
<b>Vstupné podmienky:</b>	Spustená aplikácia a pripojenie k internetu
<b>Výstupné podmienky:</b>	Vypísanie chybovej hlášky
<b>Postup:</b>	<ol style="list-style-type: none"><li>1. Používateľ zadá svoje meno a heslo.</li><li>2. Používateľ klikne na tlačidlo SIGN UP.</li><li>3. Zobrazí sa chybová hláška o existujúcom používateľovi.</li></ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 4:</b>	Akceptovanie demand
<b>Vstupné podmienky:</b>	Používateľ je prihlásený v aplikácii a má pripojenie na internet.
<b>Výstupné podmienky:</b>	Akceptovaná demand sa zobrazí vo volunteer zozname
<b>Postup:</b>	<ol style="list-style-type: none"> <li>1. Používateľ prejde na sekciu Public Demands</li> <li>2. Používateľ klikne na niektorú demand zo zoznamu</li> <li>3. Používateľ klikne na tlačidlo Accept</li> <li>4. Zobrazí sa modálne okno</li> <li>5. Používateľ klikne na tlačidlo Potvrdiť</li> <li>6. Stav požiadavky sa zmení z CREATED na ACCEPTED</li> <li>7. Požiadavka je dostupná zo sekcie volunteer</li> </ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 5:</b>	Vytvorenie demand bez itemu
<b>Vstupné podmienky:</b>	Používateľ je prihlásený v aplikácii a má pripojenie na internet.
<b>Výstupné podmienky:</b>	Zobrazí sa chybová hláška: Demand musí obsahovať aspoň jeden item
<b>Postup:</b>	<ol style="list-style-type: none"> <li>1. Používateľ prejde na sekciu Create Demand</li> <li>2. Používateľ vyplní údaje Delivery address, Expiration date</li> <li>3. Používateľ klikne na tlačidlo Create</li> <li>4. Zobrazí sa chybová hláška</li> </ol>
<b>Výsledok:</b>	PASS/FAIL



## Backend

<b>Test 1:</b>	Vytvorenie demand
<b>Vstupné podmienky:</b>	Vytvorený a prihlásený user
<b>Výstupné podmienky:</b>	Vrátený zoznam obsahuje vytvorenú demand
<b>Postup:</b>	<ol style="list-style-type: none"><li>1. Príde validná POST request na /demand</li><li>2. Vráti odpoveď 201 Created s title a id</li><li>3. Príde GET na /demand</li><li>4. Vráti odpoveď 200 Ok so zoznamom všetkých demand pre daného používateľa</li><li>5. Zoznam obsahuje vytvorenú demand</li></ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 2:</b>	Zmena profilovej fotografie
<b>Vstupné podmienky:</b>	Vytvorený používateľ
<b>Výstupné podmienky:</b>	Používateľ má v databáze nahratú fotografiu
<b>Postup:</b>	<ol style="list-style-type: none"><li>1. Príde PUT na /users/user_name.</li><li>2. Server pridá obrázok do databázy.</li><li>3. Vráti odpoveď o úspešnom pridaní obrázka.</li><li>4. Príde GET na /users/user_name.</li><li>5. Vráti údaje o používateľovi spolu s nahratým obrázkom.</li></ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 3:</b>	Akceptovanie demand
<b>Vstupné podmienky:</b>	Existujúci demand a používateľ
<b>Výstupné podmienky:</b>	Používateľ bude priradený ku demand ako volunteer
<b>Postup:</b>	<ol style="list-style-type: none"><li>1. Príde PUT na /demand/{id}.</li><li>2. Server pridá používateľa ku demand ako volunteer.</li><li>3. Vráti odpoveď o úspešnom akceptovaní demandu.</li><li>4. Príde GET na /demands/{id}</li><li>5. Server vráti detail celého demandu</li></ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 4:</b>	Vytvorenie a update demand s prázdny zoznamom items
<b>Vstupné podmienky:</b>	Prihlásený používateľ
<b>Výstupné podmienky:</b>	Vráti chybovú hlášku 400 Bad Request
<b>Postup:</b>	<ol style="list-style-type: none"> <li>1. Príde validná POST na /demand</li> <li>2. Vráti odpoveď 201 Created s title a id</li> <li>3. Príde PUT na /demand s prázdny zoznamom items</li> <li>4. Vráti 400 Bad request s textom pole items musí obsahovať aspoň jeden objekt</li> </ol>
<b>Výsledok:</b>	PASS/FAIL

<b>Test 5:</b>	Zmazanie neexistujúcej požiadavky
<b>Vstupné podmienky:</b>	Vytvorený a prihlásený používateľ
<b>Výstupné podmienky:</b>	Vráti chybovú hlášku 404 Not Found
<b>Postup:</b>	<ol style="list-style-type: none"> <li>1. Príde validná POST na /demand</li> <li>2. Vráti odpoveď 201 Created s title a id</li> <li>3. Príde DELETE na /demand/{id}</li> <li>4. Vráti odpoveď 204 No Content</li> <li>5. Príde DELETE na /demand/{id}</li> <li>6. Vráti odpoveď 404 Not Found</li> </ol>
<b>Výsledok:</b>	PASS/FAIL

# Backend

<https://github.com/xdudakm/FIIT-MTAA-2021-Backend>

## Zmeny pri REST Endpointoch oproti návrhu

Backend obsahuje aj swagger dokumentáciu dostupnú na endpointe swagger/.  
Všetky endpointy boli posunuté na api/.

### User controller

- POST signup/ zmenený na POST register/
  - response vracia aj token
- POST auth/login zmenený na POST login/
  - response vracia aj token
- POST auth/logout odstránený
- PUT users/user\_name/filename.png zmenený na PUT user/
- GET users/user\_name zmenený na GET avatar/{filename}

### Demand controller

- Jednotlivé endpointy vracajú všetky informácie dostupné ku danému demandu

### Item Controller

- Celý Item controller začlenený do Demand controller

## Postman

<https://www.postman.com/mtaa2021ds/workspace/mtaa-2021/overview>