

Adam Detko TWI

## **Nájdí part'áka na lezenie**

### **prezentácia beta verzie**

#### **Verejná inštancia projektu:**

- FE: <https://climbing-partner-fe.herokuapp.com/>
- BE: <https://climbing-partner-app.herokuapp.com/>
- databáza je na na hobby-dev pláne, takže sa občas stane, že mi zmenia prihlasovacie údaje k databáze

#### **Čo je implementované?**

- prihlásenie používateľa
- vytvorenie nového používateľa
- zobrazenie profilu používateľa (zatiaľ len seba, nie iných)
- aktualizovanie údajov používateľa (informácie a meno)
- zobrazenie všetkých oblastí
- pridanie novej oblasti pre používateľa s admin rolou
- zobrazenie detailov oblasti v ktorých sa nachádza aj zoznam všetkých ciest v rámci oblasti
- pridanie novej cesty pre používateľa s admin rolou
- zobrazenie detailov cesty aj so zoznamom používateľov, ktorí preliezli cestu
- možnosť označiť/odznačiť cestu ako prelezenú

#### **Čo je rozpracované?**

Rozpracované veci majú funkčné REST api a nie sú implementované vo FE časti

- vytváranie udalostí
- zobrazenie zoznamu udalostí (aktívnych/ neaktívnych)
- pridanie používateľa do udalosti
- zobrazenie zoznamu účastníkov udalosti
- pridávanie komentárov k udalosti
- zobrazenie zoznamu komentárov k udalosti

#### **Neimplementované časti:**

- tabuľka obtiažností ktorá sa mapuje na číselné hodnoty
- FE časti rozpracovaných vecí
- notifikácie

#### **Celkový čas implementácie: 51 hodín**

#### **Implementácia:**

Projekt má dve samostatné inštancie a databázu.

FE časť vyvíjam v reacte spolu s pomocnou knižnicou primeReact, ktorá mi poskytuje mnoho komponentov. FE komunikuje s BE pomocou REST API.

BE časť vyvíjam v Jave v Spring frameworku. Na zabezpečenie api používam spring security a na komunikáciu s postgresql databázou používam JPA.

## Záver:

Veľa vecí bolo pre mňa nových a nie všetko sa mi podarilo implementovať podľa predstáv, ako napríklad autentifikácia pomocou jwt tokenu. Tak isto na komunikáciu s REST API v reacte som si najskôr vybral useSWR knižnicu, ale po neúspešnej implementácii a nie príliš podrobnej dokumentácii som prešiel na axios knižnicu. Tiež som sa dlhšie zasekol na CORS headers, keďže mi implementácia nefunguje na globálnej úrovni. Našťastie sa to podarilo implementovať na úrovni controlerov. S niektorých vecí som sklamaný, ale nemôžem nič neodporúčať len pre to, že to neviem implementovať. Naopak by som odporúčil primeReact knižnicu na FE, a na BE by som odporúčil JPA, lebo nemusím písať natívne dotazy na databázu a dost' mi to uľahčilo robotu.

## Výstup pomocou cloc knižnice

- scanoval som git repozitár (bez súborov ktoré ignoruje git)

```
adam@adam-ThinkPad-T490:~/Documents/TWA$ cloc TWI-main.zip
 86 text files.
 82 unique files.
 10 files ignored.

github.com/AIDanial/cloc v 1.82 T=0.32 s (237.7 files/s, 99748.2 lines/s)
-----
Language               files      blank      comment      code
-----
JSON                    4           0           0          27908
TypeScript              23          137          47          1307
Java                    33          357          16          1297
Bourne Shell            1           34           62           220
DOS Batch               1           35           0           153
CSS                     10           15           5           122
Maven                   1           10           0            74
Markdown                1           20           0            26
HTML                    1            3          20            20
SVG                     1            0           0             1
-----
SUM:                    76          611          150         31128
-----
```