

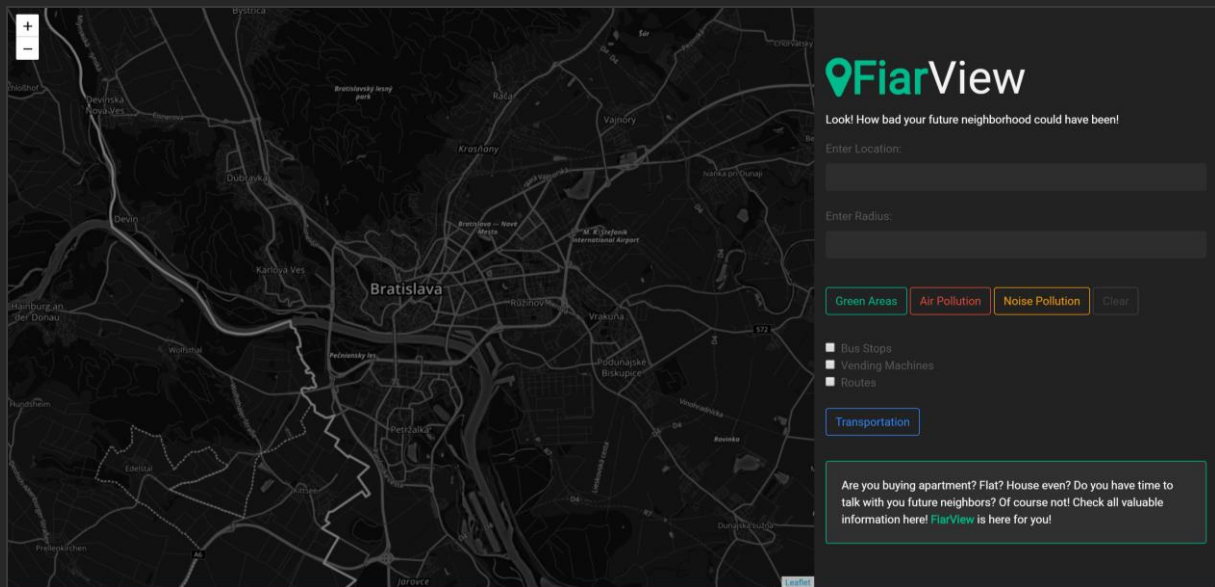


Pokročilé databázové technológie 2016/2017

Jakub Mičo

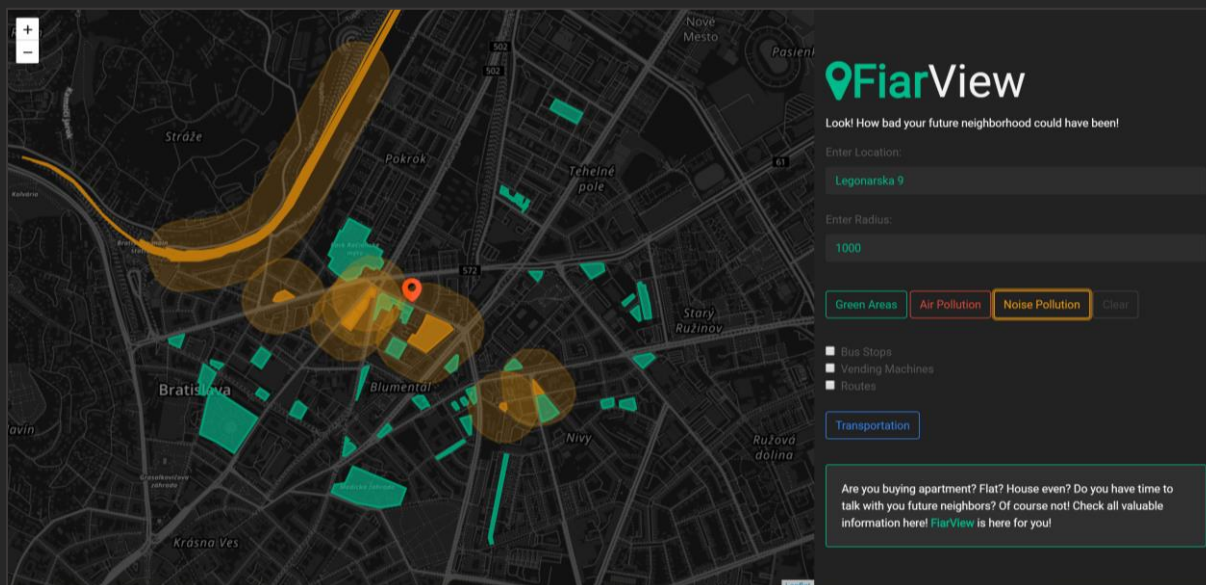
O FairView

Webová aplikácia **FairView** poskytuje prehľad okolia pre používateľom zadanú adresu. Na mape je možné zobraziť rôzne ukazovatele, ktoré môžu ovplyvniť život v danej lokalite. FairView predstavuje jednoduchý nástroj pre ľudí, ktorý zvažujú kúpu nehnuteľnosti a potrebujú sa informovať o okolí. Tak isto je vhodný pre integráciu s webovými stránkami realitných kancelárií.



Obr. 1: Landing page aplikácie

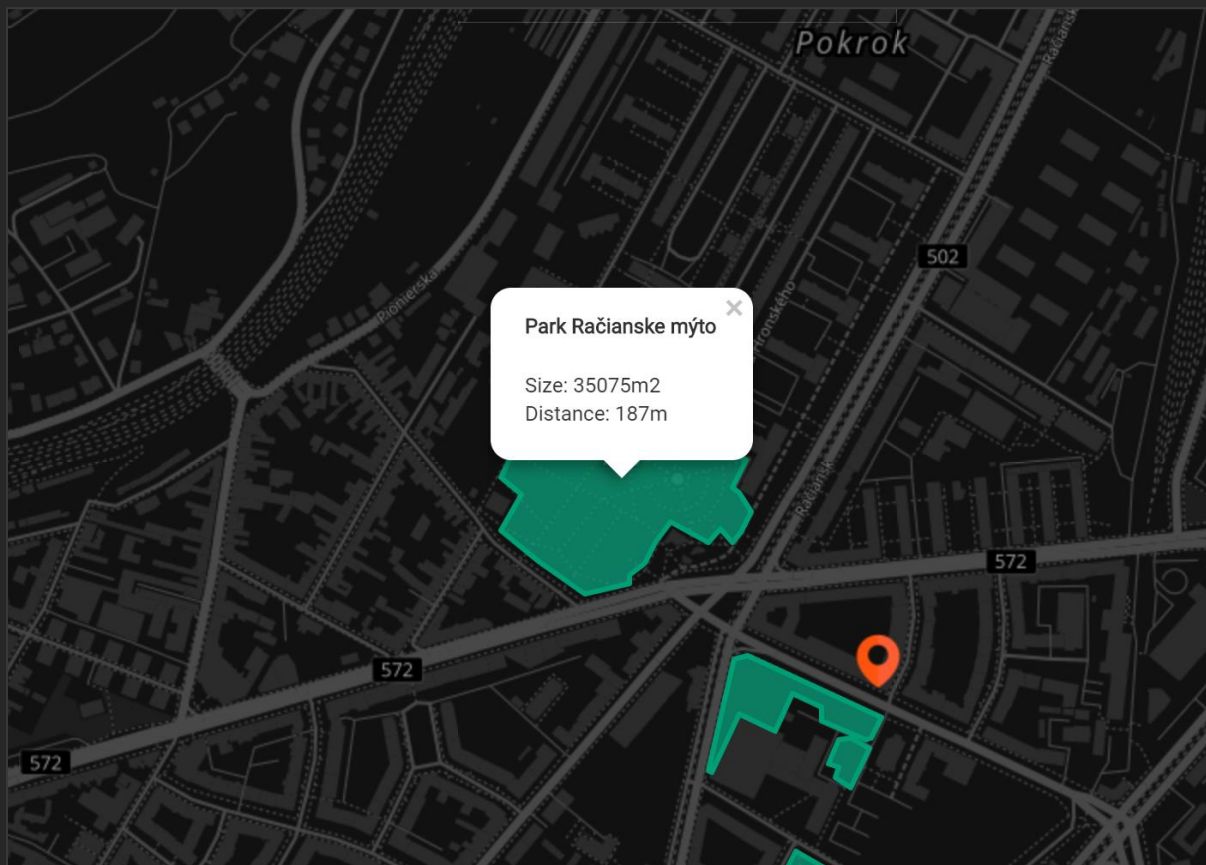
Po zadaní adresy a vzdialenosti, v ktorej chceme zobraziť výsledky, aplikácia zobrazí jednotlivé ukazovatele ako napríklad parky, znečistenie zvukové alebo vzduchu, prípadne prístup k verejnej doprave. Tieto ukazovatele sú špecifikované podľa filtra. Používateľ je povinný zadať lokáciu v podobe adresy, či mesta poprípadne ich kombináciu. Pre túto adresu sa priradí geografická poloha pomocou **Google GeoCoding API**. Následne musí používateľ špecifikovať vzdialenosť od zvoleného bodu, v ktorej chce realizovať vyhľadávanie. Tato vzdialenosť musí byť celé kladné číslo.



Obr. 2: Vizualizácia dát

Green Areas

Green Areas zobrazí všetky parky v okolí zadaného bodu, spolu s metadátami ako rozloha parku a vzdialenosť od zadanej pozície. Parky sú následne na mape vizualizované zelenou farbou.



Obr. 3: Vizualizácia metadát

Air Polution

Air polution vyhľadáva všetky priemyselné oblasti, ktoré by mohli prispieť k znečisteniu ovzdušia. Tieto zóny sú vyznačené červenou farbou.

Noise Polution

Čo bežné mapy nezobrazia, FairView áno! Hlasné zvuky ulice, železničnú trať ako aj neďalekú výstavbu, **Noise polution** zobrazí všetky zdroje zvukového znečistenia v meste spolu s ich dosahom.

Transportation

Verejná doprava a jej dostupnosť, **Transportation** zobrazí príslušné autobusové zastávky, spolu s ich traťami skrz celé mesto ako aj jednotlivé automaty na lístky.

FrontEnd

Technologie: Js, Leaflet(lib) ,Boostap(lib), Google GeoCoding

FrontEnd-ová časť aplikácie zodpovedá za komunikáciu s BackEnd-om, komunikáciu a integráciu GeoCoding API, ako aj validáciu odpovede, spracovanie a validáciu používateľských vstupov a vizualizáciu mapy a geografických dát. Funkcionalita FrontEnd-u je centralizovaná v súbore **script.js**, pozostáva z funkcií:

- getLocation() – komunikácia s Google Geocoding
- getAirPolution() – zavolá REST API s rovnakým názvom a spracuje odpoveď vo forme GeoJson
- getNoisePolution() – zavolá REST API s rovnakým názvom a spracuje odpoveď vo forme GeoJson
- getNoisePolutionRadius() – k NoisePolution získa radius, cez REST API s rovnakým názvom
- getGreenAreas() – podobne ako príklady hore
- getTransportation() – podobne ako príklady hore
- makeRequest() – vykonávanie HTTP GET requestu
- validateInputs() – validácia formulára a vstupov
- eraseData() – vymaže mapu

BackEnd

Technologie: NodeJs, Express,PQ(Npm package) PostgreSQL, PostGIS

Serverová funkcionálna je distribuovaná do súborov **app.js** a **routes.js**. App.js zabezpečuje konfiguráciu servera, ktorú môžeme vidieť na ukážke č. 1. Server je možné spustiť pomocou príkazu **npm start**. Webová aplikácia bude dostupná na adrese: <http://localhost:4200/>

```
var express = require('express');
var path = require('path');
var bodyParser = require('body-parser');
var logger = require('morgan')
var methodOverride = require('method-override');
var routes = require('./routes/router');
var app = express();

app.use("/public", express.static(path.join(__dirname, 'public')));
app.set('view engine', 'ejs');
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: false }));
app.use(methodOverride());

app.use(function (err, req, res, next) {
  console.error(err.stack)
  res.status(500).send('Something broke!')
});
app.use('/', routes);

app.listen(4200);
console.log('Starting at port 4200');
```

Ukážka 1: Konfigurácia servera

API Endpoint-y

Jednotlivé rozhrania sú uložené v súbore **route.js**, ktorý zabezpečuje validáciu adries a smerovanie.

Všetky vypublikované rozhrania zodpovedajú rovnakej štruktúre. Predstavujú koncové body, na ktoré sa dá dopytovať HTTP dotazmi pomocou metódy GET. Každá linka musí zodpovedať formátu: „<adresa>/<názovEndpointu>?lat=<latitude>&lng=<longitude>&radius=<radius>“

K dispozícií sú nasledovné API rozhrania:

- **GET** /getGreenAreas – vráti všetky parky a metadáta v zdanom okruhu zadaného bodu
- **GET** /getBusStops – vráti všetky autobusové zastávky v zdanom okruhu zadaného bodu
- **GET** /getVendingMachines– vráti všetky automaty na lítky v zdanom okruhu zadaného bodu
- **GET** /getBusRoutes – vráti všetky autobusové linky, ktoré zastanú v okolí zadaného bodu
- **GET** /getAirPolution – vráti všetky priemyselné komplexy a skládky v okolí zadaného bodu

- **GET** /getNoisePolutionRadius – vráti geometriu hlučných zón rozšírenú o radius 200 metrov
- **GET** /getNoisePolution – vráti všetky hlučné miesta(stavby, vlakové trate) v okolí bodu