

Servisno-orijentisane arhitekture

Air Monitoring

LAZAR PAVLOVIC 16268
DUSAN JANKOVIC 16110

Gateway service - Pretstavlja REST servis koji prima i prosledjuje zahteve na ostale mikroservise.

Device service - Pretstavlja mikroservis koji simulira rad senzora, koji prikuplja podatke o trenutnoj kolicini štetnih gasova(CO, SO₂, NO₂) u vazduhu i kvaliteta vazduha. Na odredjeni interval čita podatke i šalje ih data servisu. Ovaj mikroservis upravlja radom aktuatora za precišćavanje štetnih gasova.

Data service - Pretstavlja mikroservis koji prima, čuva i prosledjuje podatke koje je primio od device mikroservisa na dalju obradu. Preko ovog servisa se može pristupiti bazi podataka i interagovati sa podacima iz nje.

Notification service -Pretstavlja REST servis na kome je implementiran socket.io pub/sub sistem za asinhronu komunikaciju sa klijentom. Svaka poruka koja stigne na notification mikroservis se šalje na određeni topic i na taj način se obaveštavaju klijenti koji su subscribe-ovani na njega.

Command service -Pretstavlja mikroservis koji menja stanje aktuatora preko gateway mikroservisa.

Analytics service - Pretstavlja mikroservis koji služi za obradu podataka koje je primio od data mikroservisa i proveriti da li su u normalnim granicama. To radi tako što proverava da li je koncentracija štetnih gasova i kvalitet vazduha u zadovoljavajućoj granici. U slučaju da je neki podatak prešao odredjenu granicu šalje se zahtev command mikroservisu za promenu stanja aktuatora i šalje se poruka na notification mikroservis da bi se obavestio klijent o ograničenju.

Dashboard - Pretstavlja klijentsku angular aplikaciju koja prikazuje najnovija očitavanja senzora, podatke o senzoru i upozorenja o prevazilaženju količine štetnih gasova u vazduhu i kvaliteta vazduha.

Gateway REST

GET http://gateway:3000/stats

- uzima trenutne podatke o senzoru aktuatorima

GET http://gateway:3000/co/:id

http://gateway:3000/so2/:id

http://gateway:3000/no2/:id

- uzimaju sve podatke merenja odgovarajućeg gasa u vazduhu.

:id predstavlja id senzora(uvek 1).

GET http://gateway:3000/co/new/:id

http://gateway:3000/so2/new/:id

http://gateway:3000/no2/new/:id

- uzimaju poslednja merenja odgovarajućeg gasa u vazduhu.

:id u url-u predstavlja id senzora(uvek 1).

GET http://gateway:3000/max_value/all

- uzima vrednost granica dozvoljenih količina štetnih gasova u vazduhu.

PUT http://gateway:3000/co/cleaner

http://gateway:3000/no2/cleaner

http://gateway:3000/so2/cleaner

- menja stanja prečišćavača vazduha za odgovarajući štetni gas.

Body

```
{  
  Switch:boolean  
}
```

PUT http://gateway:3000/co/cleaner/lvl

http://gateway:3000/no2/cleaner/lvl

http://gateway:3000/so2/cleaner/lvl

- menja nivo rada prečišćavača vazduha za odgovarajući štetni gas.

Body

```
{  
  Lvl:number (u procentima 0-100)  
}
```

PUT http://gateway:3000/cleaner/all

- menja stanje svih prečisćavača vazduha i postavlja ih na određeni nivo.

Body

```
{  
  Lvl:number (u procentima 0-100)  
}
```

Notification Topics

Topics:

conection

- ispisuje u konzoli da se novi klijent subscribe-ovao na topic

warning.so2,warning.no2,warning.co

- prima upozorenje o prevelikoj količini odgovarajućeg gasa na topic

warning.aqi.low

- prima obavestenje o dobrom kvalitetu vazduha.

warning.aqi

- prima upozorenje o lošem kvalitetu vazduha.

actuator

- prima informaciju kada su actuatori promenjeni.

new.data

- prima informaciju da su novi podaci učitani.

CLI commands

docker-compose up --build - za build-ovanje i startovanje docker servisa.

cd dashboard - prebacivanje u folder gde se nalazi angular client

npm install - instaliranje potrebnih zavisnosti

ng serve --open - startovanje angular klijenta I otvaranje u browser-u.