

Dokumentacija “Flood Detection”

Projekat Flood detection je softverski proizvod koji improvizuje i simulira monitoring platformu koja prati i kontoliše par brana sa mogućnošću merenja nivoa vode, protoka vode u sekundi, i količine padavina radi predikcije i spečavanje poplave određenog područja.

Sistem za kontrolu i prikaz koristi Web dashboard koji dobija trenutne izmerene podatke sa brana, kao i upozorenje, alarme za detektovane događaje. Projekat implementira mikroservisnu arhitekturu korišćenjem različitih tehnologija za implementaciju.

Za implementaciju mikroservisa korišćen je .Net Core i Molecular framework.

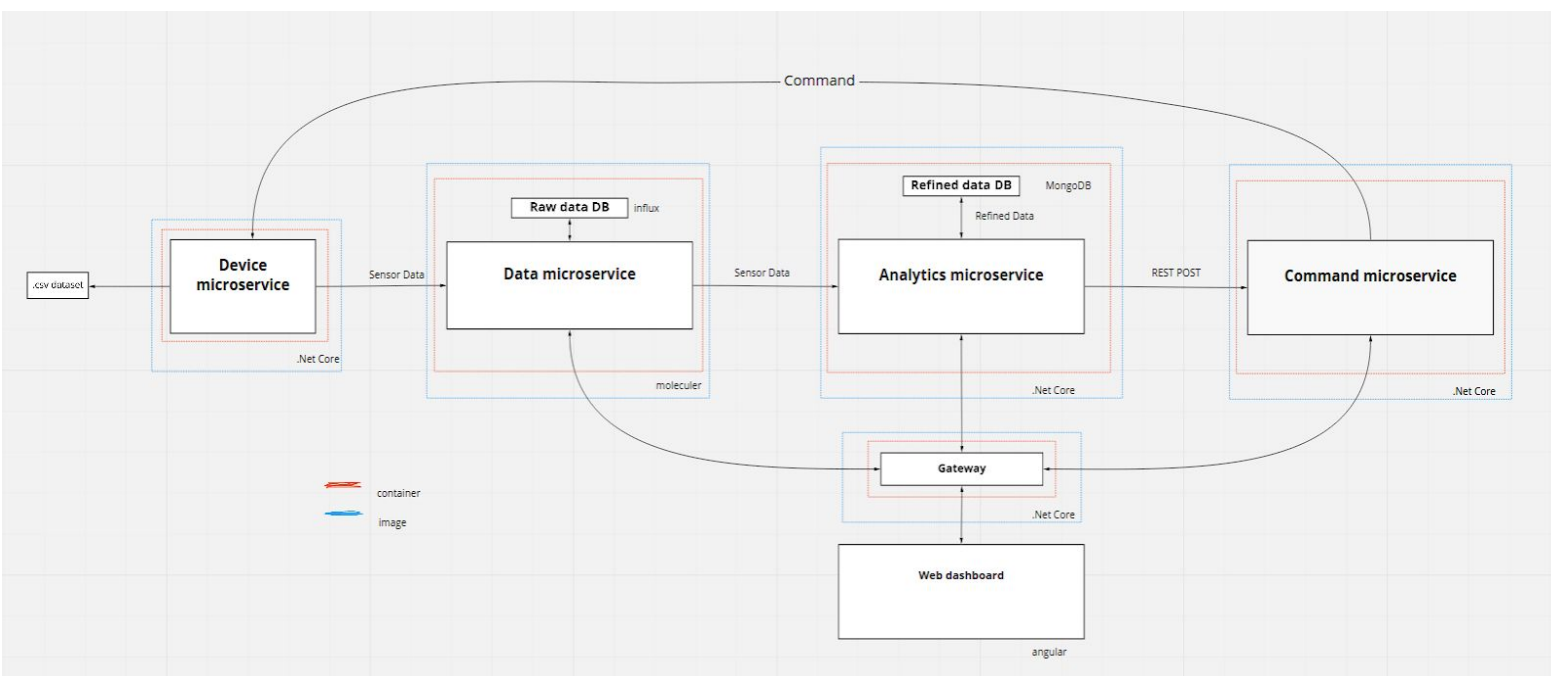
Za asinhronu komunikaciju između mikroservisa korišćen je message broker RabbitMQ.

Za perzistenciju izvornih senzorskih podataka korišćena je InfluxDB baza podataka.

Za perzistenciju analiziranih i rafiniranih podataka korišćena je MongoDB baza podataka.

Angular za Web Dashboard.

Docker za kontejnerizaciju mikroservisa.



Pregled arhitekture

Glavna stranica aplikacije gde je omogucen uvid u tok podataka sa senzora, pracenje notifikacija i mogucnost testiranja servisa.

Control panel

Data service

WaterFlow

WaterLevel

Rainfall

StationId

1

Submit

Show data

Analytics service

WaterFlow

0.5

WaterLevel

6

Rainfall

2

StationId

1

Event type

Alarm

Submit

Show data

Command service

Command name

Station Id

Flow/Level

Increase/Decrease

Ammount

Invoke

Get commands

Reset

Device service

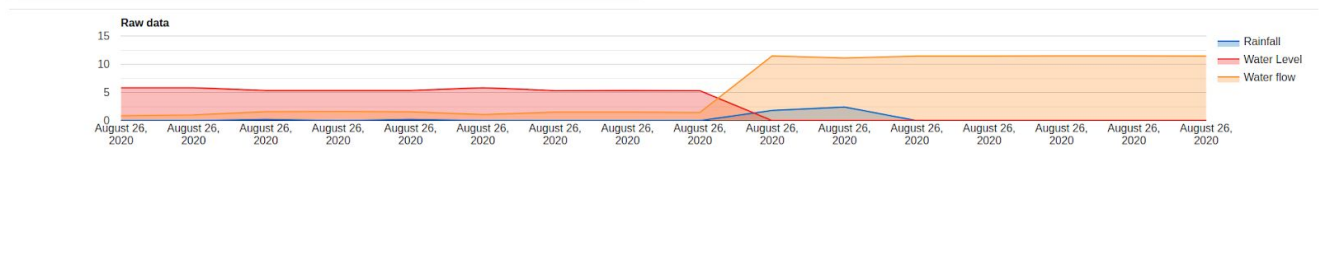
Timestep

Submit

Show metrics

Unresolved notifications

Notification: 1



Lista analiziranih podataka sa Analytics servisa tj njegove MongoDB baze podataka. Portal takodje omogucava ovakav uvid u podatke sa Device servisa tj Influx baze podataka i dostupnih komandi za Command servis.

Control panel

Data service

WaterFlow

WaterLevel

Rainfall

StationId

1

Submit

Show data

Analytics service

WaterFlow

0.5

WaterLevel

6

Rainfall

2

StationId

1

Event type

Alarm

Submit

Command service

Command name

Station Id

Flow/Level

Increase/Decrease

Ammount

Invoke

Get commands

Reset

Device service

Timestep

Submit

Show metrics

Unresolved notifications

Notification: 1

Water level	Water flow	Rainfall	Measured time	Analyzed time	Event type
5.82	1.003	0	2020-08-26T08:45:00Z	2020-09-02T20:18:54.855Z	Warn
5.821	1.062	0	2020-08-26T07:45:00Z	2020-09-02T20:19:10.83Z	Warn
5.821	0.853	0	2020-08-26T09:00:00Z	2020-09-02T20:23:12.594Z	Warn
5.821	1.062	0	2020-08-26T07:45:00Z	2020-09-02T20:23:31.626Z	Warn
6	0	2	0001-01-01T00:00:00Z	2020-09-02T20:26:15.056Z	Alarm
5.82	0.937	0.2	2020-08-25T21:00:00Z	2020-09-02T20:26:24.118Z	Warn
6	0	2	0001-01-01T00:00:00Z	2020-09-02T20:29:18.313Z	Alarm
5.82	1.003	0	2020-08-26T08:45:00Z	2020-09-02T20:33:11.026Z	Warn
5.821	1.062	0	2020-08-26T07:45:00Z	2020-09-02T20:33:26.953Z	Warn
5.82	1.003	0	2020-08-26T08:45:00Z	2020-09-02T20:35:27.482Z	Warn

Mikroservisi

1. DeviceMicroservice - .Net Core mikroservis zadužen za citanje senzorskih podataka i objavljivanje istih na SensorData topic
 - Za dobijanje trenutnih parametara(meta podataka)
 - <http://localhost:5001/getlivedata>
 - Modifikovanje vremenskog intervala očitavanja
 - <http://localhost:5001/setperiodtime>
2. Data-microservice - molecular mikroservis koji sluša na SensorData i primljene podatke čuva u Influx bazu i pritom šalje te podatke AnalyticsMicroservice-u preko REST zahteva (moguće je prebacivanje podataka i preko DataAnalytics topic-a)
 - Pribavljanje liste senzorskih podataka
 - <http://localhost:5001/raw-data>
 - Slanje senzor podataka ka data service-u
 - <http://localhost:5001/raw-data>
3. AnalyticsMicroservice - .Net Core koji prima podatke od data-microservice-a i nakon obrade ih čuva u MongoDB bazu podataka
 - Pribavljanje liste obradjenih podataka
 - <http://localhost:5001/getrefineddata>
 - Prijem senzorskih podataka na obradu
 - <http://localhost:5001/newsensordata>
4. CommandMicroservice - Mikroservis implementiran framework-om .Net Core, obezbeđuje REST API za dobijanje dostupnih komandi, i pozive dostupnih komandi.
 - Dobijanje liste dostupnih komandi i njenih parametara
 - <http://localhost:5001/getcommands>
 - Post endpoint-i:
 - <http://localhost:5001/decreasewaterlevel>
 - <http://localhost:5001/increasewaterlevel>
 - <http://localhost:5001/decreasewaterflow>
 - <http://localhost:5001/increasewaterflow>
 - <http://localhost:5001/reset>
5. Gateway - .Net Core servis koji omogućava pristup microservice-ima u aplikaciji. Implementacija se zasniva na tome da se na osnovu poznatih ruta (u okviru routes.json) odradi preusmeravanje zahteva koji je pristigao na Gateway ka svom pravom odredistu. U slučaju prosirivanja API-ja nekog servisa dovoljno je dodati novu rutu u okviru routes.json fajla.

Asinhrona komunikacija

Projekat koristi RabbitMq message broker za komunikaciju između mikroservisa i to:

1. Dva topic-a za prenos izvornih senzorskih podataka :
 - SensorData na koji se prijavljuju zainteresovani Data mikroservis.
 - DataAnalytics na koji se prijavljuje zainteresovani Analytics mikroservis.
2. Jedan topic za prenos komandi na koji se prijavljuje zainteresovani Device mikroservis: Command