Proiect SD

-documentatie-

Energy management online platform

By: Andrei Dumitrescu

1. Obiectiv

Aplicatia are ca obiectiv usurarea monitorizarii diferitelor device-uri de masurare a electricitatii. Utilizatorul isi poate crea un cont, vizualiza device-urile pe care le detine, consumul de curent la momentul vizionarii si consumul de curent istoric. Fiecare device masoara curentul cu ajutorul unui senzor. Un utilizator poate avea mai multe device-uri de masurare, in timp ce un device poate avea un singur senzor.

Adminul aplicatiei poate introduce noi utilizatori / device-uri / senzori, sa le vizualizeze, sa le stearga sau sa le updateze. Tot odata, acesta poate mapa device-uri la utilizatori si senzori la device-uri.

1. Cerinte Functionale

Admin-ul poate sa:

* Se logheze / delogheze
* Aplice operatii de tip CRUD pe utilizatori
* Aplice operatii de tip CRUD pe device-uri
* Aplice operatii de tip CRUD pe senzori
* Mapeze device-uri catre utilizatori
* Mapeze senzori catre device-uri

Utilizatorul poate sa:

* Se logheze / delogheze
* Vizualizeze device-urile pe care le detine
* Vizualizeze senzorii device-urilor
* Vizualizeze curentul consumat de acestea la momentul acesarii aplicatiei
* Vizualizeze media curentului consumat de acestea din momentul in care au fost asignate acestuia
* Vizualizeze un grafic

1. Cerinte non-functionale

Securitatea aplicatiei este cerinta non-functionala a acestui proiect. Pentru a asigura acest lucru, utilizatorul trebuie sa se logheze cu credentialele pe care si le-a create in urma inregistrarii. Toate tipurile de conturi create din aplicatie sunt de tipul regular user. Administratorii se introduc manual, printr-un query in baza de date. Un utilizator nu poate accesa pagina administratorilor ( folosind protected routes si json web tokens ), cat nici un administrator nu poate accesa pagina unui utilizator.

1. Cazuri de utilizare

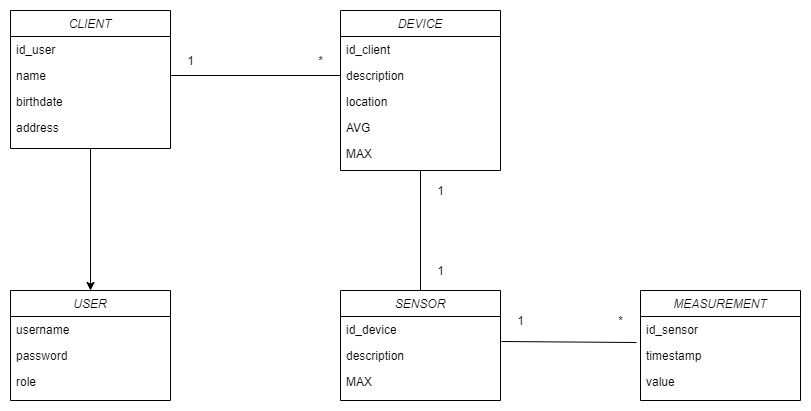
Utilizator:

1. Inregistrare utilizator – utilizatorul completeaza formularul de inregistrare de pe home page
2. Vizualizarea datelor despre device-uri si senzori – utilizatorul se logheaza de pe pagina de home page, este redirectionat catre pagina de utilizator, acolo unde poate vedea datele despre device-urile si senzorii sai
3. Delogare – utilizatorul apasa pe butonul de Log-out si este redirectionat la homepage, unde-si poate crea un nou cont sau sa se logheze din nou

Admin:

1. Vizualizarea utilizatorilor / device-urilor / senzorilor – administratorul se logheaza, alege una dintre cele 3 optiuni din meniul dropdown “Menu” - Users / Devices / Sensors, iar mai apoi este redirectionat la pagina unde se pot vedea tabelele ce contin respective optiune aleasa
2. Introducerea de utilizatori / device-uri / senzori – dupa ce isi alege obiectul pe care vrea sa-l adauge si este redirectionat la pagina respective, acesta apasa pe butonul “Add … “ (User / Device / Senzor). Trebuie respectate niste reguli in legatura cu introducerea anumitelor campuri, care se pot vedea atunci cand se editeaza respectivul camp.
3. Editarea de utilizatori / device-uri / senzori – administratorul apasa pe butonul de ‘Edit’ in tabelul ales inainte, in dreptul coloanei pe care vrea s-o editeze. Field-urile se vor transforma in input-text-uri, care se pot edita. Diferite mesaje de eroare pot aparea, in cazul in care nu sunt respectate anumite reguli, moment in care editarea nu se face, iar admin-ul trebuie sa reinceapa procesul de editare. In momentul in care termina, apasa pe butonul ‘Done’. In cazul editarii cu success, un mesaj de confirmare va aparea.
4. Stergerea de utilizatori / device-uri / senzori – administratorul apasa pe butonul de ‘Delete’ in tabelul ales inainte, in dreptul coloanei pe care vrea s-o stearga. Diferite mesaje de eroare pot aparea, in cazul in care nu sunt respectate anumite reguli. In cazul stergerii cu success, un mesaj de confirmare va aparea.
5. Maparea clientilor la device-uri – administratorul va alege din meniu sectiunea de ‘Map Devices to Clients’, unde va vedea toti clientii si toate device-urile(impreuna cu utilizatorii mapati la ele – daca au un utilizator mapat, acestea nu mai pot fi mapate la alti utilizatori). Pentru a realiza maparea, se introduce numele utilizatorului in primul input si numele device-ului in cel de-al doilea. Diferite mesaje de eroare pot aparea. In cazul in care maparea se face cu success, un mesaj de confirmare va aparea.
6. Asocierea device-urilor la senzori – administratorul va alege din meniu sectiunea de ‘Associate Devices to Sensors’, unde va vedea toate device-urile si toti senzorii(impreuna cu device-urile associate la ele – daca au un device asociat, acestea nu mai pot fi associate la alti senzori – acest lucru se aplica si pentru device-uri – un singur senzor poate fi mapat la un device, nu mai multe, ca in cazul utilizatorilor, de mai sus). Diferite mesaje de eroare pot aparea. In cazul in care maparea se face cu success, un mesaj de confirmare va aparea.

5.Diagrama simplificata a bazei de date



6.Conectarea aplicatiei la simulatorul de masuri ale senzorilor folosin RabbitMQ

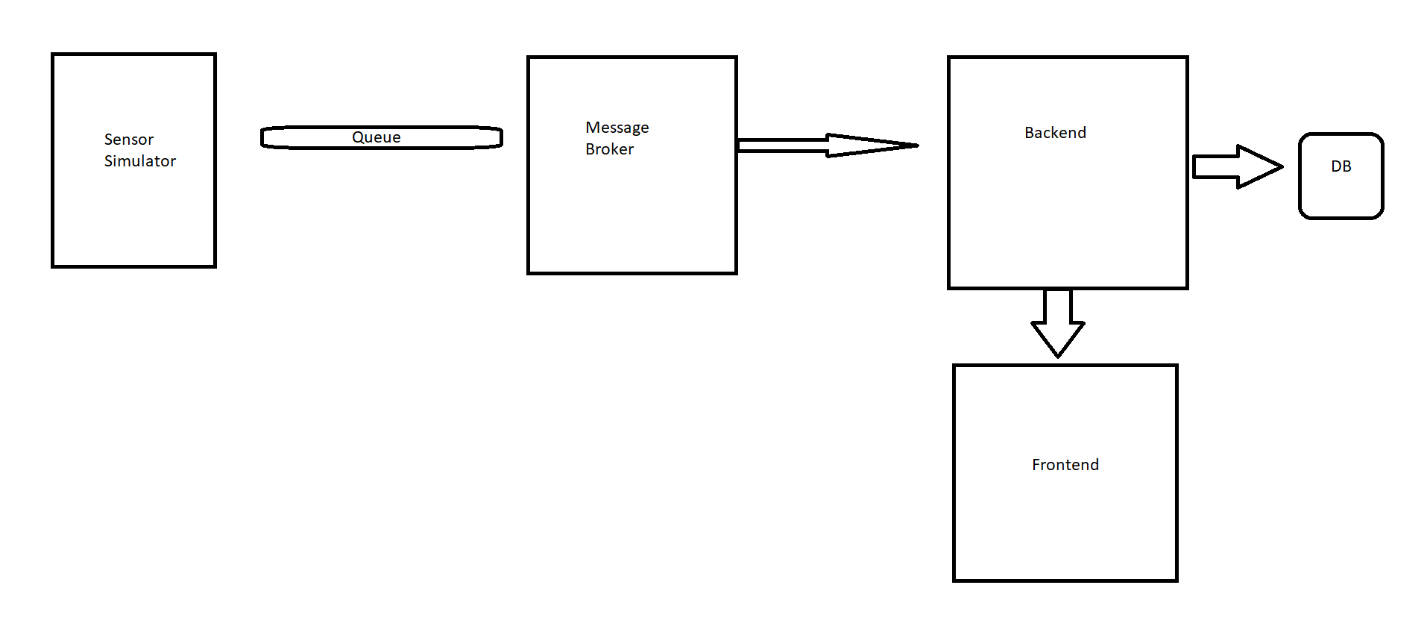
Conform temei 2, am realizat un simulator de masuratori, care extrage datele dintr-un fisier ce contine valori care reprezinta valoarea masurata de un senzor(sensor.csv) odata la 10 minute. Pentru a seta senzorul pentru care este generata valoarea, avem un fisier denumit “config.txt” in care putem seta ID-ul senzorului pentru care citim valoarea din sensor.csv.

Odata la 10 minute, programul citeste ID-ul si datele din fisierul .csv, genereaza un timestamp si trimite datele intr-o coada, implementata cu ajutorul tool-ului numit RabbitMQ.

Coada ramane in permanenta deschisa, iar in momentul in care apare un mesaj in ea, un broker consuma mesajul transmis in coada. Acesta afiseaza mesajul primit din coada, iar mai apoi transmite printr-un request de tipul POST catre backend, mesajul primit (care contine id-ul, timestamp-ul si masuratoarea senzorului).

In backend se afiseaza, in consola, mesajul primit.

Mai jos se poate observa arhitectura intregii aplicatii, impreuna cu modulele de Sensor-Simulator si Message-Broker, specifice temei 2:



7.Introducerea endpointului RPC pentru graficul masuratorilor

Conform temei 3, am realizat o aplicatie web, in care utilizatorul isi poate urmarii, pe un grafic, media masuratorilor de la o anumita ora, facute de un anumit senzor.

Senzorul este setat de catre programator, la cerinta clientului. Astfel, pentru fiecare zi separata, pe grafic, se va vedea o linie de culoare diferita care arata indica masuratorilor facute de senzor pentru o anumita ora (Practic, se face media tuturor masuratorilor de la ora 13:00-13:59 si se vor afisa, pe grafic, la ora 13 s.a.m.d pentru restul orelor).

Pentru implementarea acestui sistem, s-a folosit RPC in partea de backend. Backend-ul expune un endpoint de tip WebSocket. Acest endpoint este ‘lovit’ prin call-ul aplicatiei de frontend, trimitandu-se ID-ul senzorului pentru care vrem sa primim date.

Tipul de RPC folosit a fost json-rpc 2.0. Astfel, datele pe care clientul (frontendul) le primeste, arata de forma {id: ‘id’, medianPerDayAndMonthAndHour: ‘medianPerDayAndMonthAndHour’, result: {status: ‘success}}.

Datele efective care il intereseaza pe client(frontend) sunt medianPerDayAndMonthAndHour. Formarea acestui array se face prin parsarea datelor relevante primite din tabela de monitored\_values. Acesta este un array care contine informatii despre valoarea medie de energie consumata intr-un interval orar, pentru o anumita zi si pentru o anumita luna. Mai departe, in client (frontend), aceste date sunt parsate pe zile si luni, iar mai apoi transmise catre componenta Chart, care este afista pe ecran.

Mai jos se poate observa noua arhitectura a aplicatiei intregi, updatata, in care s-a adaugat si modulul de frontend pentru graficul mediei pe zile si ore, folosindu-se RPC :

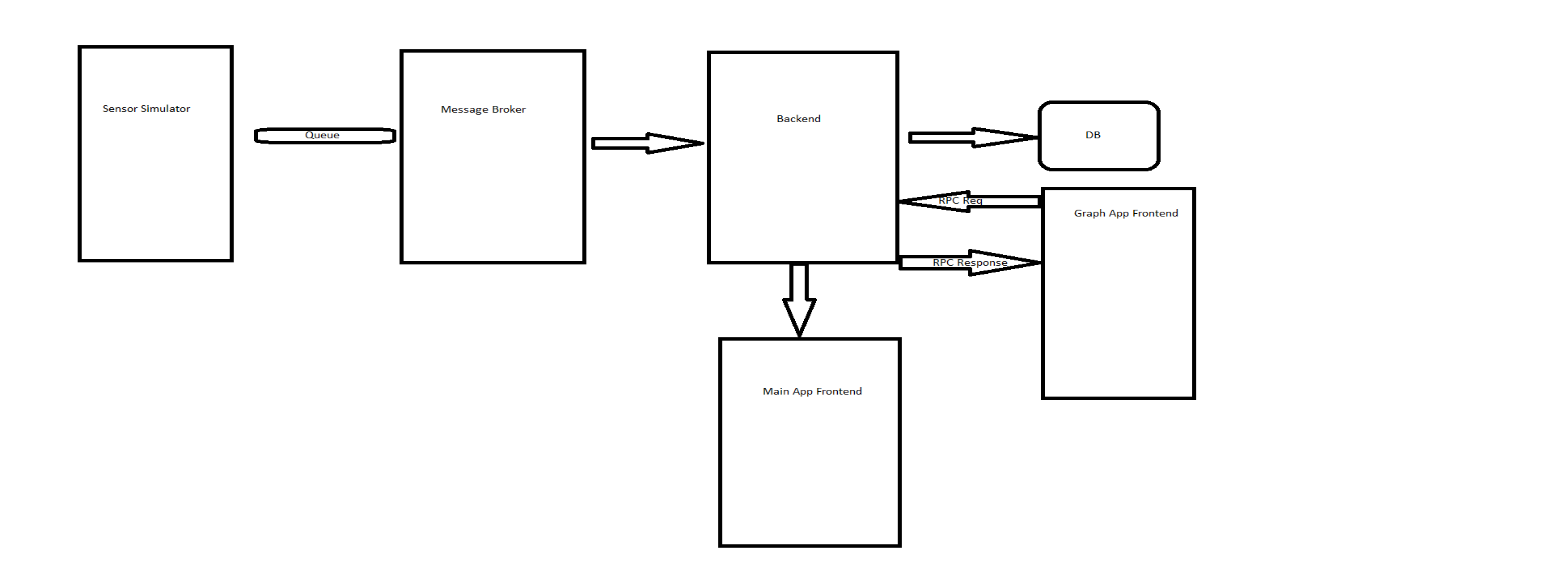


Diagrama de deploy se poate observa mai jos:

