Documentatie- Buretia Alex-Dian

1.Context

Microserviciile sunt programare care impart aplicatia mare si care , spre deosebire de un monolit, permit a face face debug mai usor decat pe un monolit, dar aceste microservicii trebuie sa comunice unele cu celelalte pentru a avea acelasi rezultat.

2. Prezentarea temei

Aceasta tema a constat dintr-o aplicatie de manageruit dispozitivele de catre utilizatori. Un adiminsitratir poate sa adauge, sa editeze, sa stearga sis a citeasca din baza de date un dispozitiv (acest lucru se face prin componentele acestui dispozitiv: id, adresa, descriere,consumul maxim si id-ul contului care il detine. Observam ca intre cont si dispozitiv exista o relatie de 1 la N( un cont poate detine mai multe dispositive, dar un dispozitiv poate fi a unui singur cont). Administratorul poate sa faca aceste operatii si pe baza de date a utilizatorilor, prin modificarea campurilor specific acestuia(id-ul, numele, rolul sau : poate fi ADMIN sau CLIENT precum si lista cu device-uri). Prin adaugarea de persoane si device-uri, se va face maparea dintre acestea. Datele sunt introduce de catre administrator astfel. In plus, daca utilizatorul isi va edita un dispozitiv care ii apartine, aceasta schimba se va propaga in baza de date a dispozitivelor prin microserviciul de conturi care comunica cu cel de dispozitive.

3. Descrierea proiectului

Back-end-ul consta din 2 microservicii care comunica in cadrul unei metode si sunt descrise in Java cu biblioteca Spring Boot si plugin-ul Maven.

In cadrul miroservicului de client am folosit o clasa pentru controller, in cadrul careia am mapat porturile si am executat operatiile de de CRUD pe baza de date de la client prin metodele specific. (“/accounts”) este adresa mapata din cadrul portului 8091 de care se foloseste aplicatia mea. Metoda updateAccountDevice este metoda care primeste de la front-end noul dispozitiv editat si il tramsite la back-end-ul microservicului de device-uri pentru a fi salvat. Astfel, aceasta metoda foloseste un URL pentru a tine cont de faptul ca aplicatia se afla intr-un container si localhost nu reprezinta adresa acestuia. Portul microservicului de device-uri este 172.19.0.5 , adresa IP pe care am trecut-o in locul localhost. Metoda postForEntity apeleaza metoda POST din microservicului de device-uri si salveaza cu succes device-ul editat de noi. Ca paramaetru se trimite device-ul. Clasa AccountDto este reposnsabila cu primirea de la front-end a datelor, care nu sunt bagate automat in enitate deoarece clientii nu ar trebui sa aibe acces la informatii sensible. Dto-ul contine exact aceleasi campuri ca si entitatea dar fara adnotarile care contruiesc tabela din baza de date. In folderol de modele, am inclus clasele Device si Account, deoarece primeste de la front-end un device si il trimite la celalalt back-end. In plus, am adaugat si un repository pentru a facilita accesul la metodele importante din JPARepository. Clasa Config este pentru CORS si pentru a putea conecta back-end-ul la front-end. Enum-ul RoleA descrie rolurile pe care le poate avea un utilizator(de admin sau client). In application.properties se efectueaza legarea la baza de date specifica. Clase asemanatoarea se afla in microserviciul de device-uri. Acest microserviciu se conecteaza la alt port (8091) cu alta baza de date .

Front-end-ul este implementat in React si consta din 4 pagini : una pentru admin si pentru toate operatiile implementate de el(adaugarea, editare, stergere si vizualizare de device-uri, precum si de utilizatori), alta pentru a putea clientul sa vizulizeze device-urile si sa le editeze daca e necesar (acest lucru este posibil printr-un tabel care contine aceste detalii despre dispositive si un buton de Edit care ne deschide un modal in pagina noastra), una pentur home. La operatiile de adaugare a unui device sau utilizator, se va deschide un modal care ne permite sa introducem toate detaliile stipulate si sa le salvam. In cazul in care dorim sa salvam un dispozitiv pentru un utilizator, putem sa alegem dintr-un multiselection dropdown checklist si sa le selectam pe toate cele dorite. In cazul adaugarii la device a unui utilizator, se va deschide un single select dropdown. In cazul editarii acestor date, se va deschide acelasi modal dar butonul pentru a realiza aceasta operatie este cel de ‘’Edit”. Modalul apartine astfel de pagina care l-a deschis dar este individuala, spre deosbire de modalul de la editarea clientului, care apare direct in pagina. Modalul care ne permite sa editam device-uri este diferit de cel care ne permite sa editam conturile.

Deployment diagram:

