Temă de proiect

Obiectiv:

Acest proiect își propune să demonstreze utilitatea conceptelor învățate la cursul de Arhitectura Sistemelor Software.

Proiectul este un exercițiu practic ce constă în proiectarea și implementarea unei aplicații de microservicii combinată cu arhitectura de tip monolit distribuit și cea bazată pe evenimente utilizând un API Gateway, protocoale de comunicație REST, gRPC și un broker de mesaje Kafka. Construcția acestei aplicații va fi facilitată și de folosirea unui framework front-end precum Svelte dar și de o soluție de containerizare precum Docker.

Rezultatul va fi o pagină web ce returna informații despre un client sau o companie în funcție de opțiunile alese de utilizatorul aplicației.

!!!! Proiectele copiate sau asemănătoare din punct de vedere al codului scris si al abordării nu vor fi luate in considerare si vor primi nota 1.

Detalii de implementare:

Pentru implementare, se vor avea in vedere următoarele aspecte:

- Înainte de a putea accesa orice resursă, utilizatorul trebuie sa se autentifice folosind un server separat cu una din tehnologiile uzuale folosite în acest moment precum API Keys, OAuth, JWT, LDAP, etc.
- Pagina de start va permite utilizatorului să introducă datele necesare căutării:
  - Nume client: input de tip text
  - Tipul informației: client sau companie, opțiune selectabilă dintr-un meniu dropdown
  - Obține informații: butonul ce declanșează căutarea
  - Vizual, pagina de start va conține elementele de căutare din Fig. 1

Fig.1 – Elementele din formularul de căutare

- Deoarece obiectivul aplicației este unul strict demonstrativ de arhitectură, nu va fi necesară implementarea unei baze de date sau a unei structuri de date complexe. Cele două înregistrări prezentate în scenariile de mai jos sunt suficiente.
- Arhitectura sistemului software proiectat va trebui să respecte conținutul din Diagrama 1

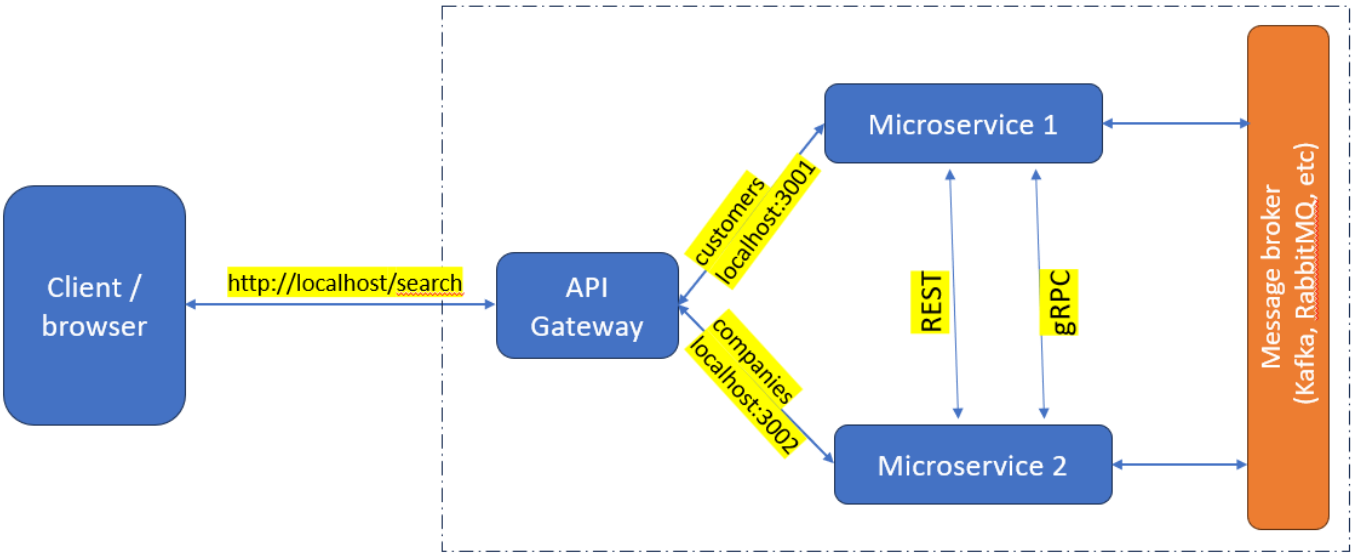

Diagrama. 1 – Arhitectura aplicației

Detalii de funcționare

În funcție de tipul căutării selectate de utilizatorul aplicației, vom avea următoarele trei scenarii:

Scenariul 1:

Clientul introduce un nume valid de client, selectează opțiunea *Client* din meniul *Tip* și apasă butonul *Obtinere informatii.*

Scenariul 1

Din acest moment interogarea va avea următorul parcurs:

1. Va pleca spre API gateway cu endpoint-ul http://localhost/search
2. API gateway va redirecționa cererea către adresa http://localhost:3001 (Microserviciul 1).
3. Pentru a răspunde, Microserviciul 1 va apela prin protocol REST un endpoint de la Microserviciul 2, care va răspunde cu informația:
   - Avere detinuta: $USD 10,000,000
4. După ce răspunde către Microserviciul 1, Microserviciul 2 va trimite prin brokerul de mesaje pe un canal la care Microserviciul 1 este abonat, următorul mesaj:
   - Client:<nume client>; Functie in companie:CEO

5. După ce va aduna cele două rânduri de informație, Microserviciul 1 va răspunde către client cu informația necesară afișării în pagina principală.

Scenariul 2:

Din browser – clientul introduce un nume valid de companie, selectează opțiunea *Companie* din meniul ***Tip*** și apasă butonul ***Obținere informații***.

Scenariul 2

Din acest moment interogarea va avea următorul parcurs:

1. Va pleca spre API gateway cu endpoint-ul http://localhost/search
2. API gateway va redirecționa cererea către adresa http://localhost:3002 (Microserviciul 2).
3. Pentru a răspunde, Microserviciul 2 va apela prin protocol gRPC un endpoint de la Microserviciul 1, care va răspunde cu informația:
   - Valoare estimata: $USD 70,000,000
4. După ce răspunde către Microserviciul 2, Microserviciul 1 va trimite prin brokerul de mesaje pe un canal la care Microserviciul 2 este abonat, următorul mesaj:
   - Companie:<nume companie>; Numar de angajati:25,000
5. După ce va aduna cele două rânduri de informație, Microserviciul 2 va răspunde către client cu informația necesară afișării în pagina principală.

Scenariul 3

Din browser – clientul introduce un nume invalid de client sau companie, selectează opțiunea din meniul ***Tip*** și apasă butonul ***Obtinere informatii***.

Pe pagina principală se va afișa informația: Nu s-a găsit niciun rezultat în baza de date.

Scenariul 3

- Modalitatea prin care se va implementa scenariul trei va fi decisă de proiectant.

Alte aspecte de implementare:

- Cele 5 componente principale ce alcătuiesc sistemul (interfața client, API gateway, Microserviciul 1, Microserviciul 2, Broker de mesaje) vor fi încapsulate într-un container de tip Docker.

Barem de punctaj:

Punctajul oferit pentru proiect va fi acordat după cum urmează:

- Un punct din oficiu (1p)
- Implementarea corectă a serviciului de autentificare (1p)
- Interfața front-end funcțională (0.75p) și containerizată (0.25p)
- Înfrumusețare interfață front-end prin aplicarea unui framework CSS și adăugare de fluiditate (0.5p)
- API gateway funcțional (0.75p) și containerizat (0.25p)
- Microserviciul 1 - funcțional (0.75p) și containerizat (0.25p)
- Microserviciul 2 - funcțional (0.75p) și containerizat (0.25p)
- Broker de mesaje funcțional, integrat cu aplicația și containerizat (1 p)
- Comunicație corectă REST între Microserviciul 1 și Microserviciul 2 – (0.75p)
- Comunicație corectă prin broker de mesaje între Microserviciul 1 și Microserviciul 2 – (0.5p)
- Comunicație corectă gRPC între Microserviciul 2 și Microserviciul 1 – (0.75p)
- Comunicație corectă prin broker de mesaje între Microserviciul 2 și Microserviciul 1 – (0.5p)