

Site de licitații

1. Introducere

1.1 Descrierea temei

Am creat o aplicație web de licitații a unor produse. Aceasta va avea o simplă și ușor de utilizat de către utilizatori. Pentru crearea acestei aplicații m-am inspirat din modelul unei aplicații deja existente și anume Ebay.

Aplicația este realizată în Angular pentru interfațata utilizatorului iar pe partea de backend am utilizat limbajul Java, datele sunt stocate în PostgreSQL, acesta este o baza de date relațională.

2. Scop & Obiective & Rezultate

2.1 Scop

Posibilitatea de a licita un produs atât timp cât utilizatorul este logat iar tranzacția respectă normele legale.

2.2 Obiective

- 1 sistem de autorizare și de autentificare bazat pe token.
- O interfață de utilizator.
- Opțiunea de a adăuga o poză la produs.
- Implementarea unui pub/sub pentru actualizarea prețurilor.

2.3 Rezultate

3. Cerințele proiectului

- Identificarea actorilor și a cazurilor de utilizare
- Aplicația va dispune 3 tipuri de utilizatori:

- 1.Utilizator nelogat(Fig.1);
- 2.Utilizator logat(Fig.2);
3. Admin(Fig.3);

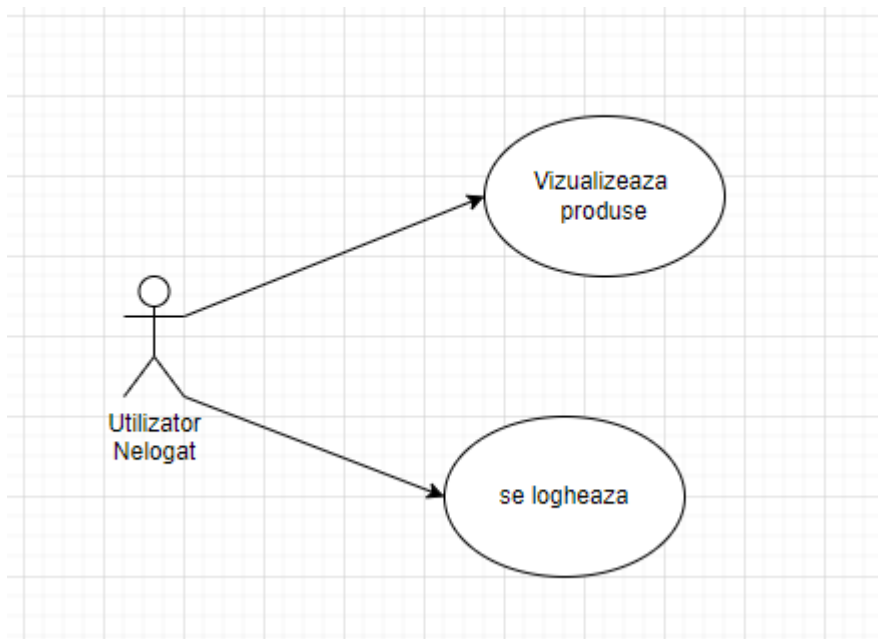


Fig.1

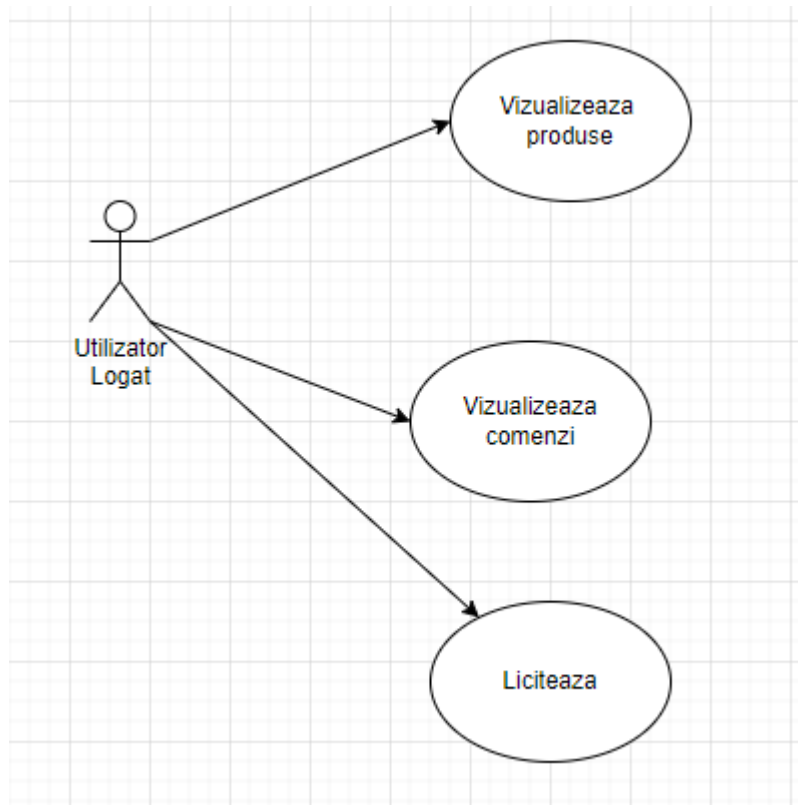


Fig.2

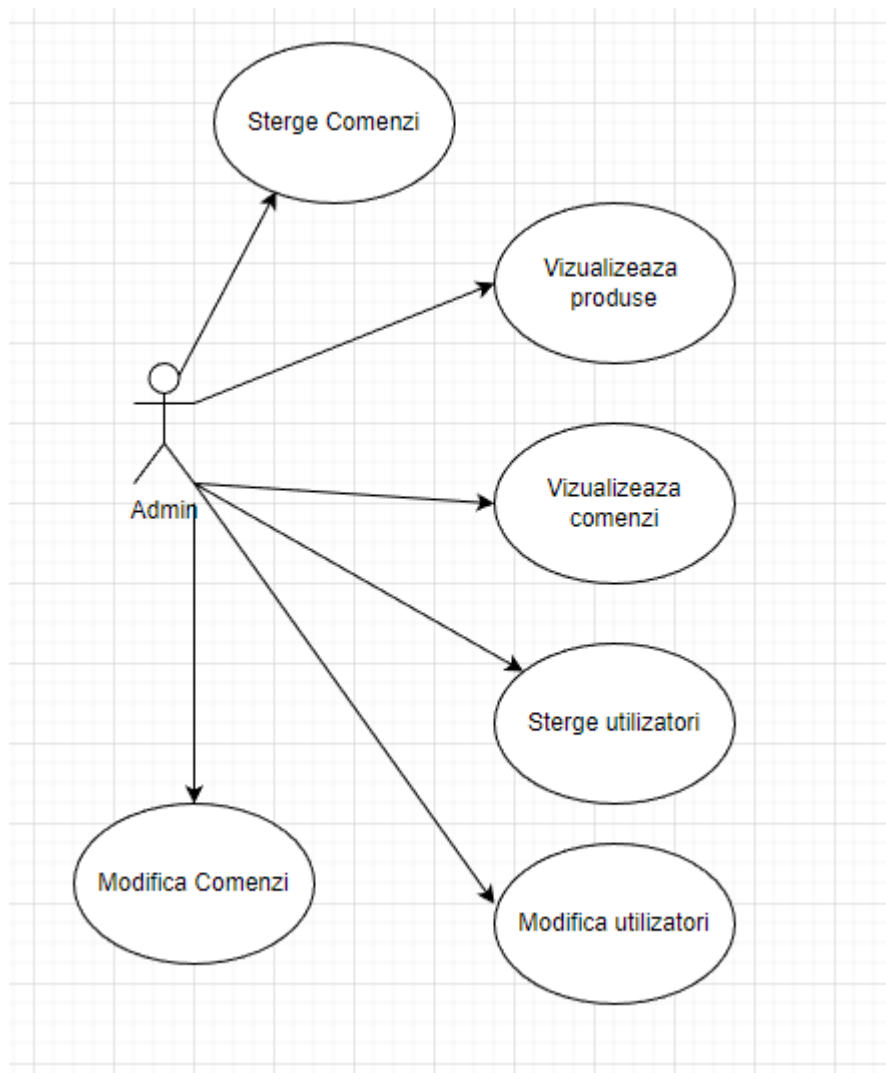


Fig.3

• Cerinte functionale

- Aplicatia va actualiza pretul in timp real folosind un sistem distribuit
- Aplicatia se va folosi de Kafka pentru a trimite mesaje livratorului
- In cazul unui update, aplicatia se va actualiza automat fara permisiunea utilizatorului, pentru a evita neplacerile unui pop-up si a mentine siguranta si performanta aplicatiei

• Cerinte non-functionale

– Utilizabilitate

1. In interiorul aplicatiei va exista un link catre un ghid online de folosire pentru utilizatorii noi;
2. Interfața se va axa pe nuanțe de vișiniu si gri.
 - Performanta 1. Aplicația va fi disponibila online pe laptop sau PC;
 - 2. Aplicația va avea o reacție de răspuns de maxim 6 secunde;
 - 3. Sistemul poate prelua date 22/24h și va actualiza baza de date.

– Fiabilitate

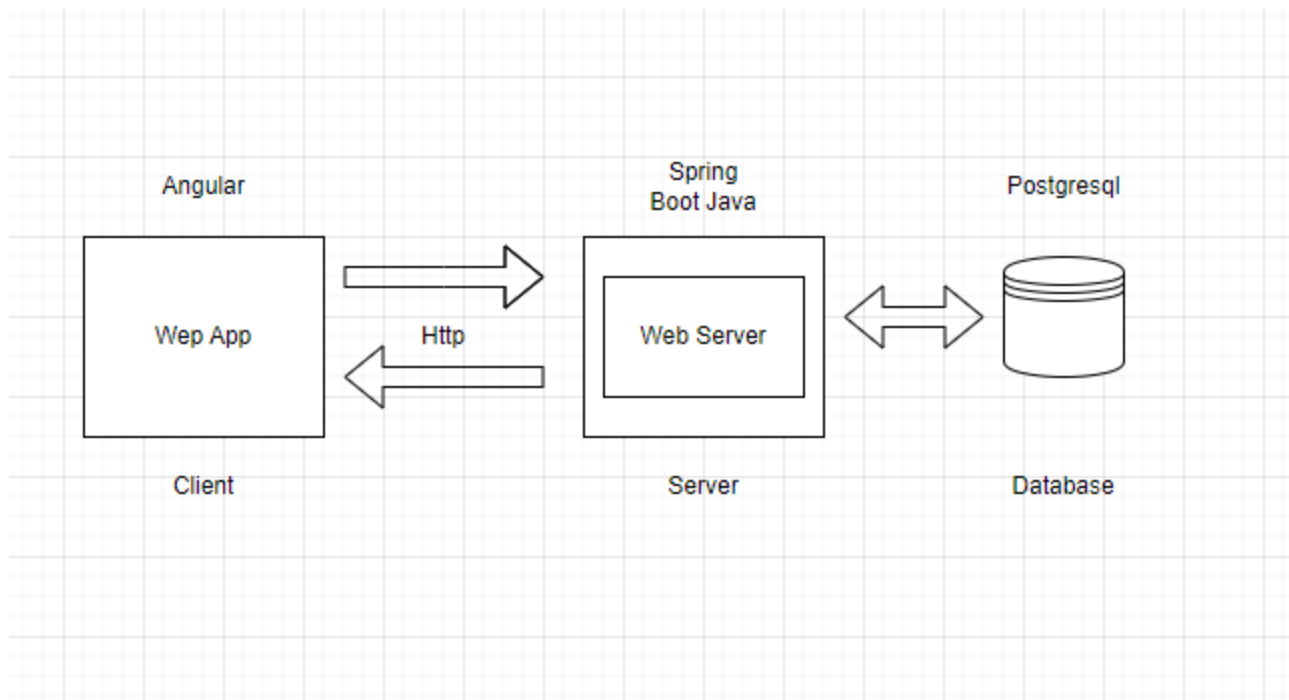
1. Aplicația va funcționa cat timp comanda de Run este activa;
2. In cazul unor intrări invalide, se va afișa un mesaj de eroare.

– Implementare

1. Baza de date: PostgreSQL;
2. Back-end: Java
3. Front-end: Angular;
5. Mediu de dezvoltare: IntelliJ.

4.Arhitectura

Arhitectura rest, folosind Java pe partea de backend si Angular pe partea de frontend, postgresql ca baza de date.



5.Design

6. Metodologia de implementare

Pentru implementarea proiectului a fost aleasa metodologia Agile.

1. Primul sprint(2 săptămâni) am lucrat la arhitectura proiectului și am creat baza de date. Am decis să merg pe o structură de tip n tier layer.
2. Al 2-lea sprint(2 săptămâni) am lucrat la partea de autentificare și autorizare.
3. Al 3-lea sprint(2 săptămâni) am lucrat la design ul aplicației.
4. Al 4-lea sprint(2 săptămâni) am creat commands si

queries pe baza de date.

5. Al 5-lea sprint(2 săptămâni) am lucrat pe implementarea unui sistem distribuit(kafka).

7. Proiectare și detalii de implementare

- Baza de date

Baza de date conține 4 tabele: BoughtProduct, ExpiredProduct, Product, People(deoarece user e keyword in PostgreSQL)