

Soni Shop

10.01.2019

Sontica Alexandru-Florentin

Soni Shop S.R.L.

Sector 1, Calea Victoriei 31

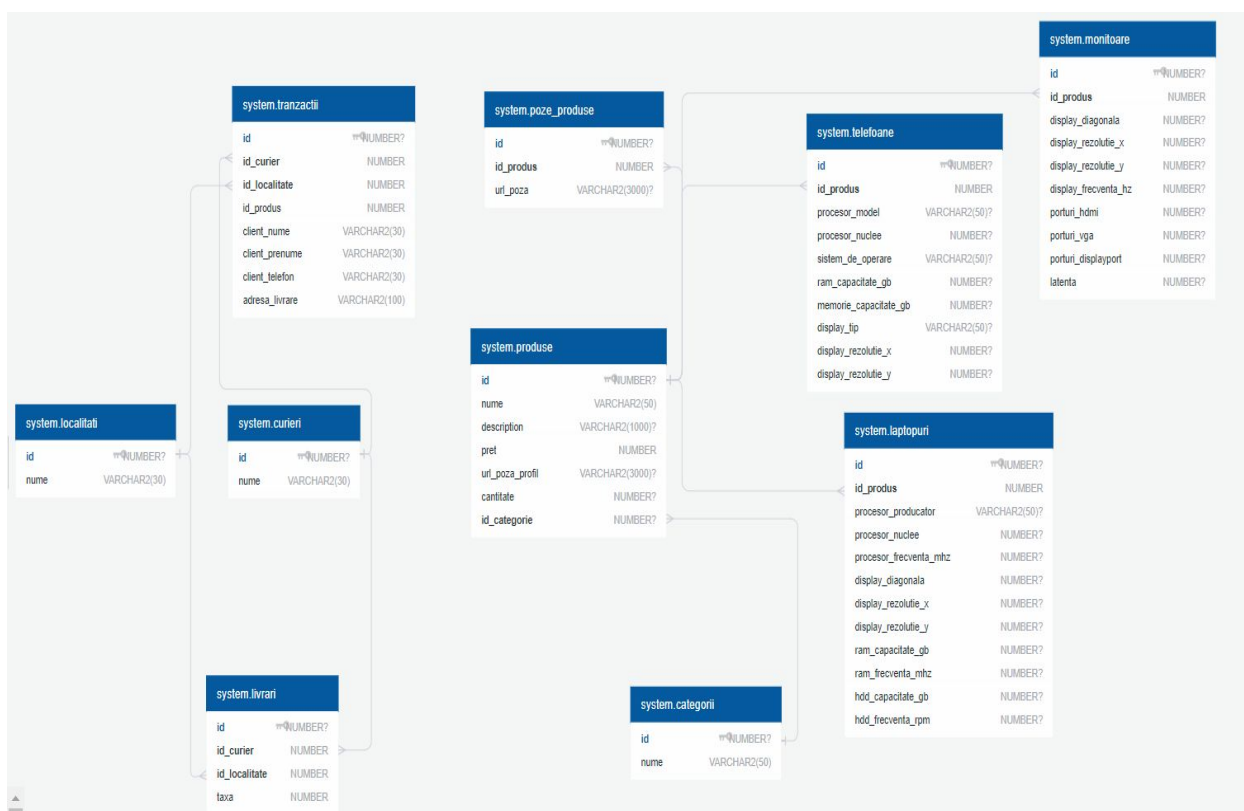
Bucuresti

Descrierea temei

Am ales sa fac un site de ecommerce specializat pe tehnologie. Cele 3 componente mari ale aplicatiei sunt: frontendul (implementat in frameworkul React), backendul (mai exact, servicii REST implementate in frameworkul Spring) si baza de date (Oracle Express Edition 11g - instalata in docker).

Descrierea bazei de date

1. Diagrama bazei de date



2. Structura tabelor

Pentru partea aplicatiei care se ocupa de plasarea comenzilor (selectare localitate, selectare curier, introducere adresa/nume/prenume/telefon si trimiterea comenzii) avem 4 tabele:

Localitati care, de fapt, sunt cele 42 de judete ale Romaniei

Curieri DHL, Fan si UPC ca exemple

Livrari tabel aditional pentru rezolvarea relatiei multi-multi intre localitati si curieri; mai exact, ne putem da seama din el de toate localitatile in care livreaza un curier, si de toti curierii care livreaza intr-o anumita localitate

Tranzactii tabel ce memoreaza o comanda plasata. Pentru fiecare comanda, vom stoca curierul, localitatea, produsul si informatiile clientului

Pentru produsele disponibile pe site

Produse este un tabel generic care contine informatii generice despre toate categoriile de produse de pe site (id, nume, descriere, prete, poza profil, stoc disponibil)

Poze este un tabel de contine un numar arbitrar de poze pentru fiecare produs. Tabelul este adaugat pentru rezolvarea relatiei one-to-many intre produs si pozele acestuia

Urmeaza tabelele ce reprezinta categorii explicite de produse

Laptopuri (procesor, display, ram etc)

Telefoane (procesor, sistem de operare, memorie etc)

Monitoare (diagonala, porturi, latentă etc)

3. Descrierea constrangerilor de integritate

Pentru partea aplicatiei care se ocupa de plasarea comenzilor, toate coloanele (mai putin cheile primare) au constrangeri de integritate NOT NULL. Asta e deoarece nu vrem sa avem curieri fara nume, sau judete fara nume, sau sa nu stim ce taxa percepe curierul UPC in judetul Alba. In plus, in tabelele livrari si tranzactii, avem constrangeri de FOREIGN KEY pe coloanele id_curier si id_localitate (referentiaza id din tabela Curieri si id din tabela Localitate), intrucat nu vrem sa adaugam inregistrari in cele 2 tabele cu curieri sau localitati inexistenti

Cat despre tabelele Poze, Laptopuri, Telefoane si Monitoare, toate au constrangere de integritate FOREIGN KEY pe id_produs (care referentiaza id din tabela Produse)

4. Descrierea procedurilor si functiilor

Toate procedurile si functiile sunt implementate in pachetul comenzi.

Functii

Functia adauga_tranzactie este cea care insereaza o noua comanda in tabela de tranzactii. Ea returneaza (intai catre backend, iar apoi backendul catre frontend) id noii comenzi plasate, pentru a putea fi urmarita ulterior de catre user.

Dupa insertul facut de aceasta functie in tabela tranzactii, se declanseaza triggerul insert_tranzactii (este trigger after insert on system.tranzactii).

Triggerul insert_tranzactii se duce in tabela Produse, ia elementul cu id = id_produs din tranzactii, si scade stocul disponibil al produsului cu 1.

Proceduri

Obține_curieri: primește numele unui județ și returnează un cursor cu toți curierii care livrează în acel județ

Obține_localitati: returnează un cursor cu toate județele din România

Obține_categorii: returnează un cursor cu toate categoriile disponibile (la început, doar laptopuri, telefoane, monitoare)

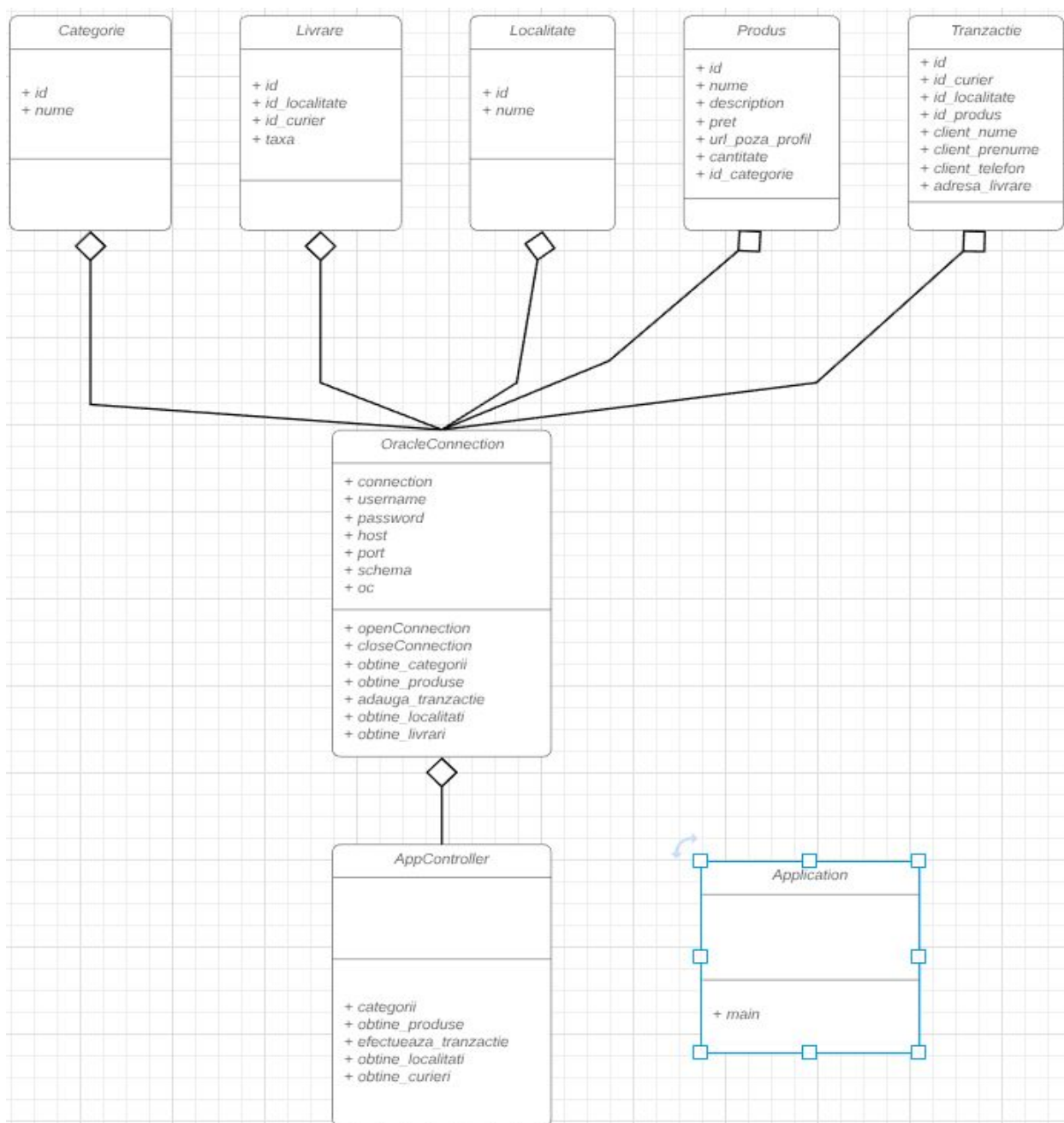
Obține_laptopuri: returnează informații generice despre toate laptopurile (prin cursor)

Obține_telefoane: returnează informații generice despre toate telefoanele (prin cursor)

Obține_monitoare: returnează informații generice despre toate monitoarele (prin cursor)

Descrierea aplicatiei

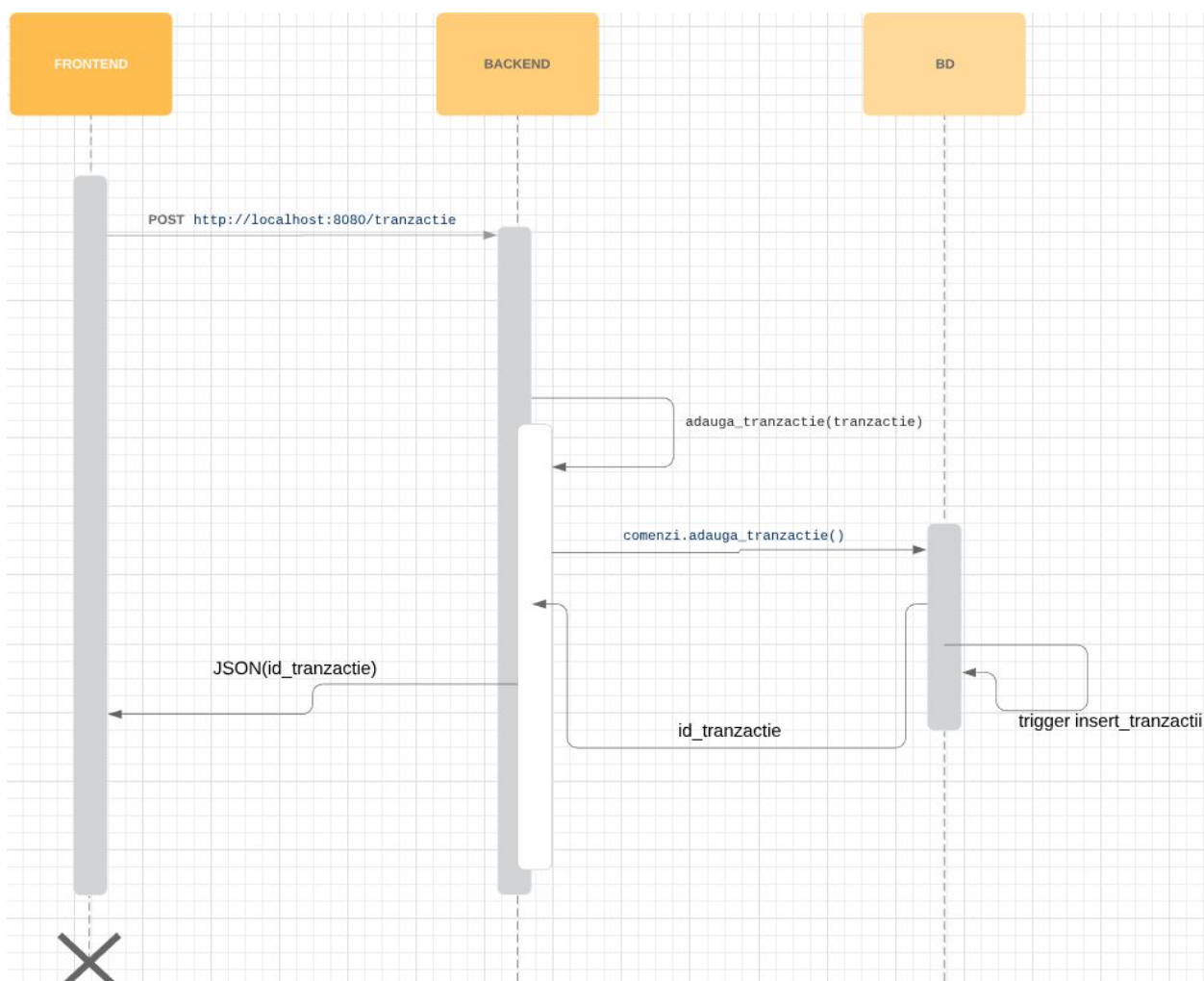
1. Diagrama de clase

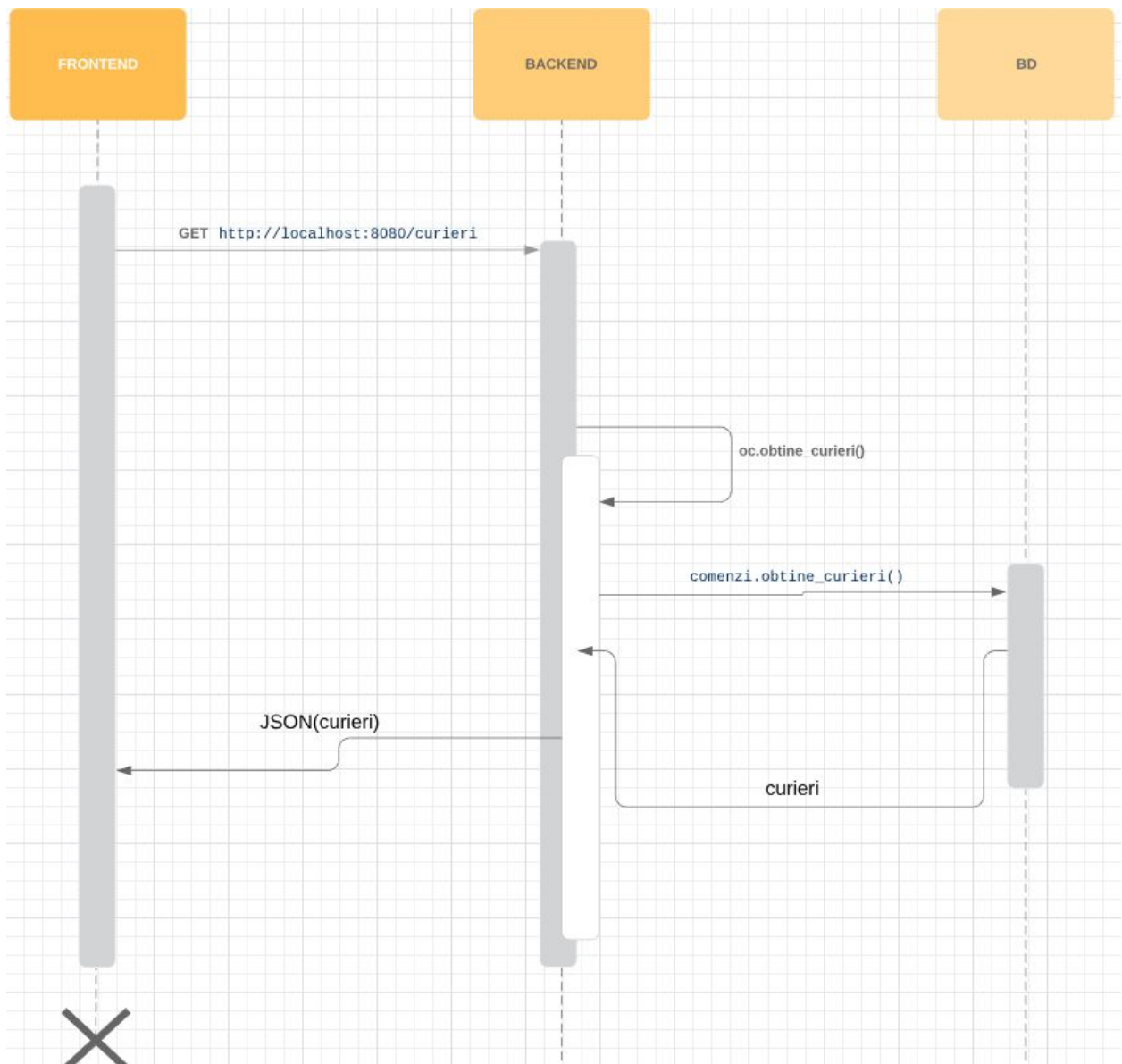


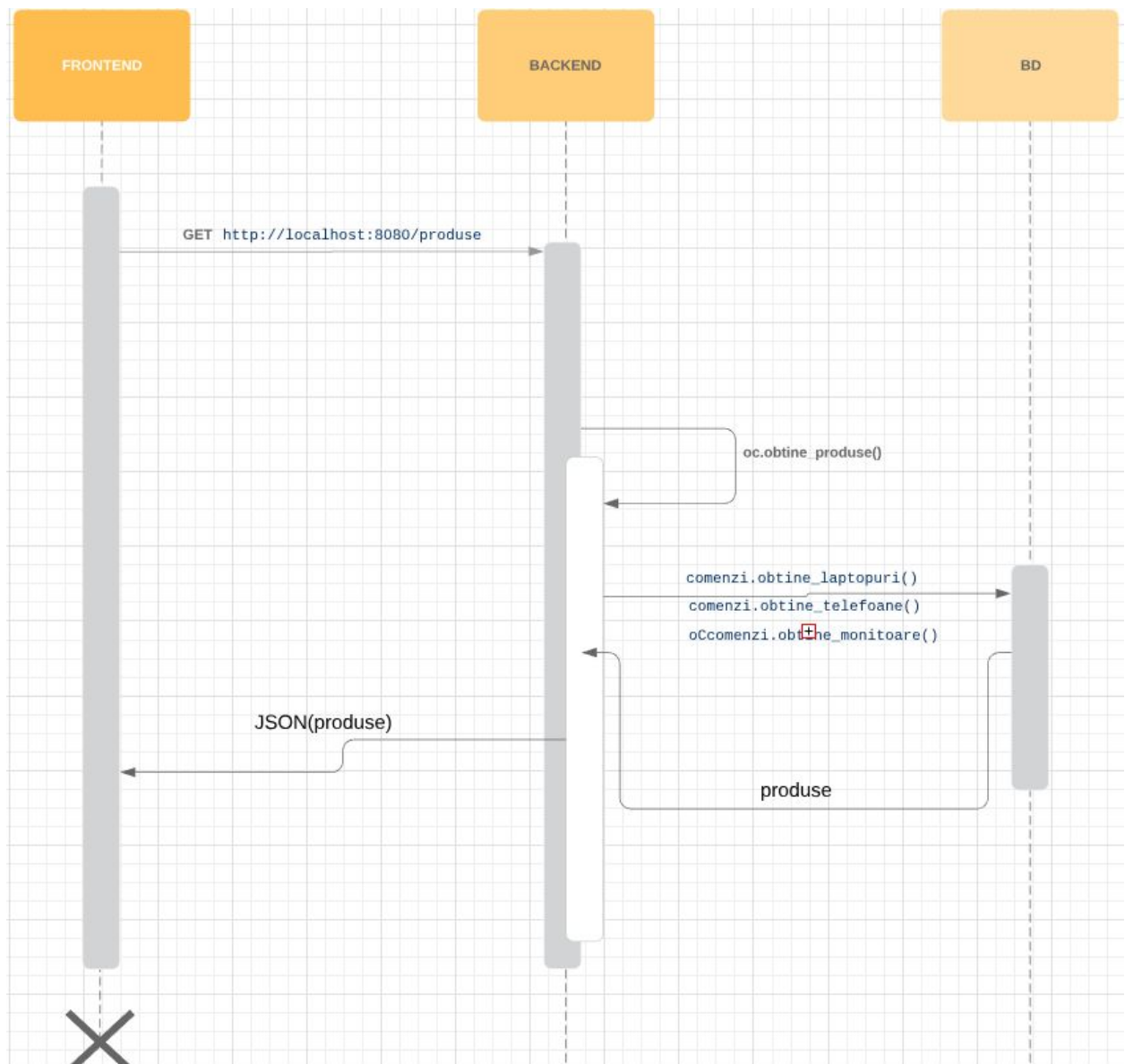
2. Structura claselor

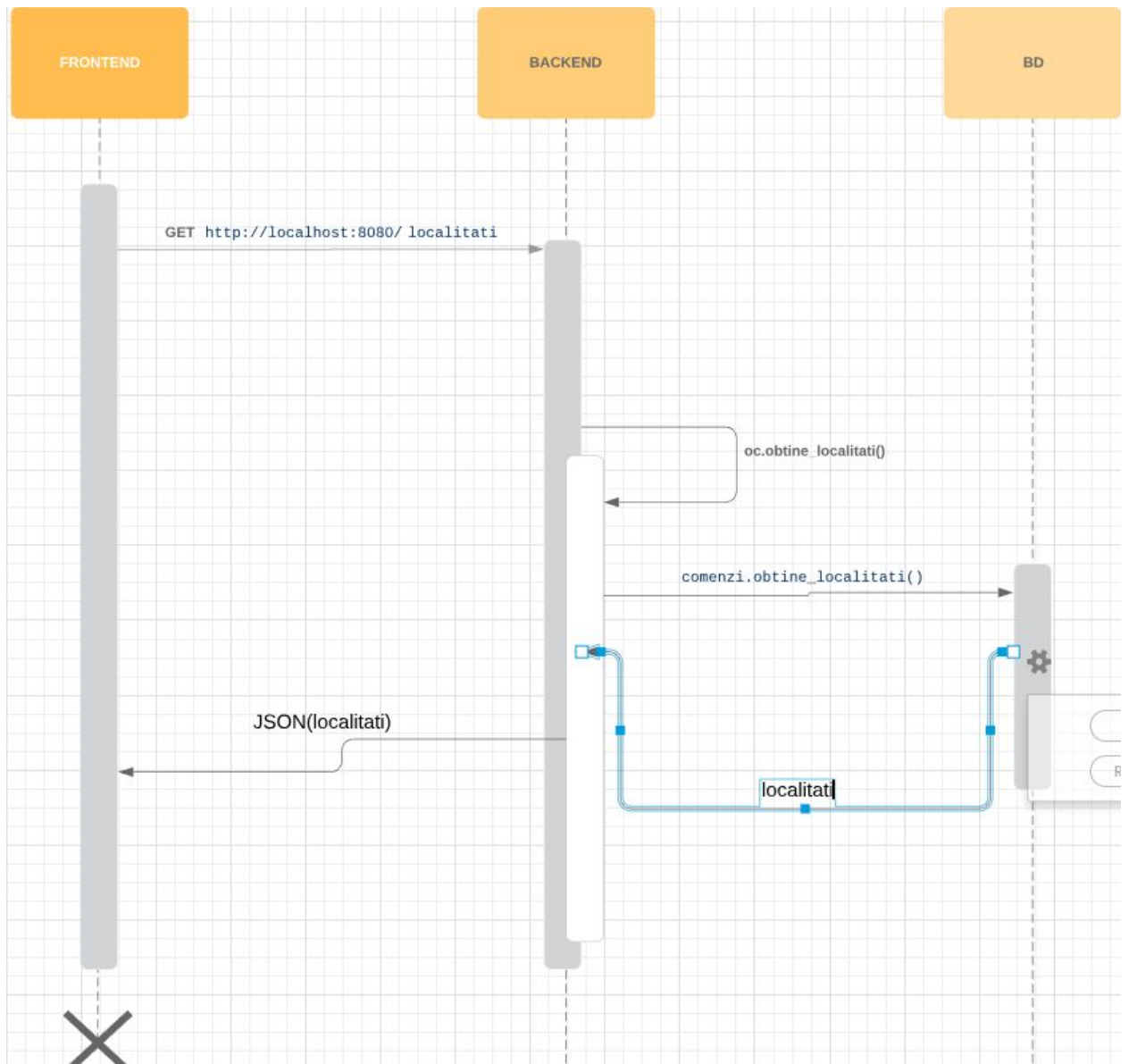
Toata logica incepe din ApplicationController, care e clasa ce implementeaza toate serviciile REST de care se foloseste frontendul. De fiecare data cand se primeste un REST call, ne folosim de un OracleConnection static per clasa, setat anterior (in main), pentru a a apela metodele specifice. Metodele din OracleConnection fac apeluri de functii si proceduri catre baza de date. Clasele folosite si returnate de OracleConnection pentru a modela baza de date sunt: Categorie, Librare, Produs, Localitate, Tranzactie.

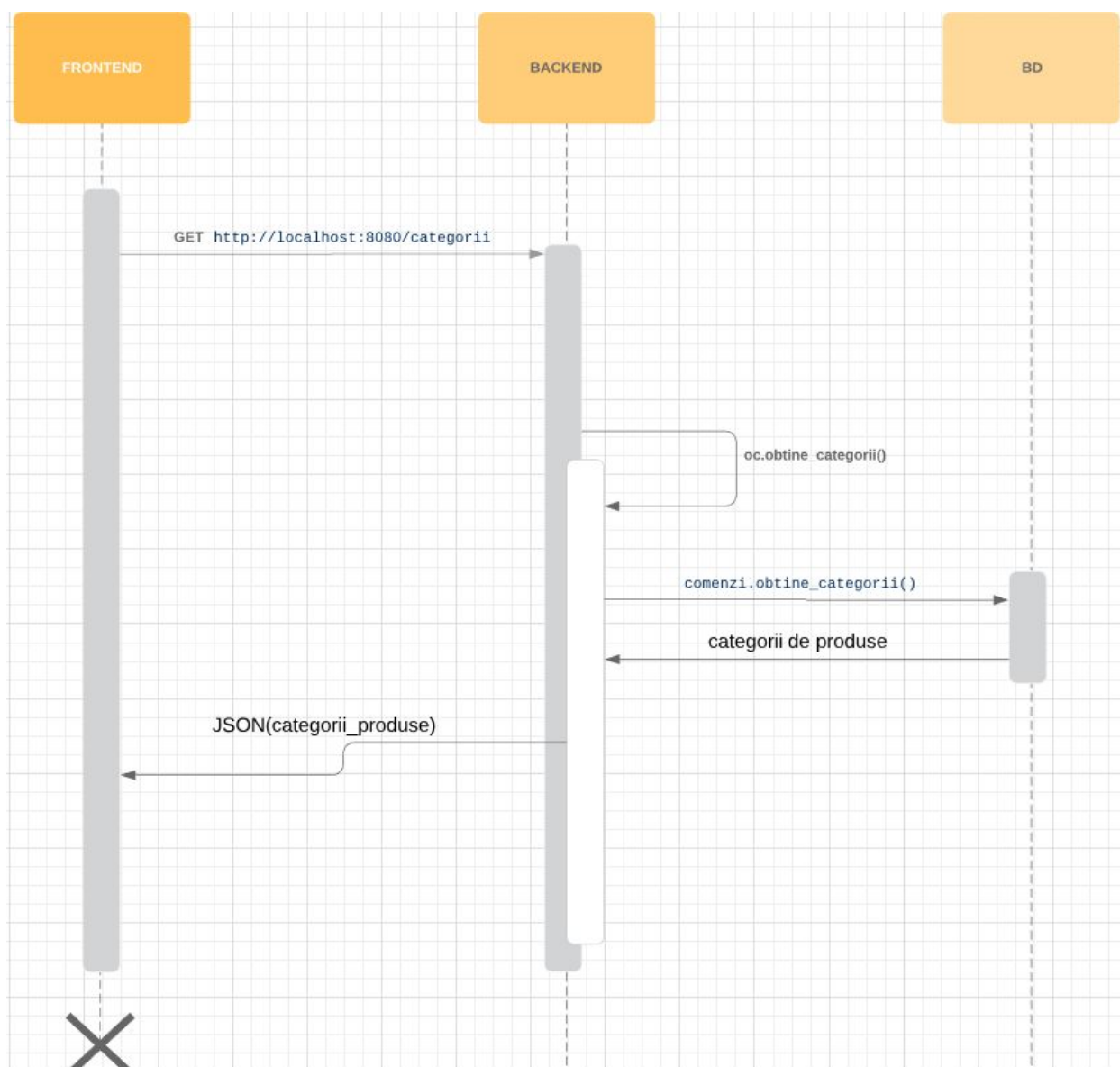
3. Diagrama de stari si fluxul de lucru al aplicatiei











4. Prezentarea modului in care se face conexiunea cu baza de date

Conexiunea cu baza de date se face ca in pdf-ul din arhiva incarcata pe cs.curs, denumita Database Connection. Este facuta pe baza unui `OracleConnection`. In el, se incarca clasa `oracle.jdbc` si se deschide o conexiune catre baza de date.

Capturi de ecran pentru interfete si rapoarte

Bun venit la Soni Shop!



Bun venit la Soni Shop!



Bun venit la Soni Shop!

Categorii de Produse

- laptopuri
- telefoane
- monitoare

LED Samsung 24



CUMPARA

Bun venit la Soni Shop!

Categorii de Produse

- laptopuri
- telefoane
- monitoare

Nume

Prenume

Telefon

Adresa

Judet

Curieri

Categorii de Produse

- laptopuri
- telefoane
- monitoare

Nume

Prenume

Telefon

Adresa

Judet

Curieri

localhost:3000 says
Ati lasat campuri necomplete!

OK

localhost:3000 says
Comanda dumneavoastra are id-ul 1

OK

Categorii de Produse

laptopuri
telefoane
monitoare

Nume Mihaila
Prenume Popescu
Telefon 0770481601
Adresa Strada Victoriei 12
Judet Dolj
Curieri Fan
Submit

Concluzii

A fost un proiect provocator, din care am avut de invatat foarte multe. Am reusit sa imi expandez cunostintele in toate cele 3 mari componente ale unei aplicatii web: frontend, backend si database.

Bibliografie

www.spring.io

www.reactjs.org

<https://docs.oracle.com/en/database/>