Bontida are din nou nevoie de ajutor

Student: Gabor George Cătălin
Grupa: 30239

April 22, 2018

Cuprins

1 Analiza cerințelor

1.1 Specificare cerinte

Se cere realizarea unei aplicaiți web pentru prezentarea unui magazin pentru fectivalul Electric Castle. Acesta va avea două tipuri de utilizatori

- administrator;
- user normal.

fiecare avand diferite privilegii. Nu se cere o implementare pentru un sistem de login.

1.2 Cerinte functionale

Aplicația trebuie să pună la dispoziție utilizatorilor următoarele funcționalități:

- toate operațile făcute de un utilizator vor fi marcate prin tipul său in URL (ex: pentru admin, URL-ul va avea o formă aseamănatoare cu http://localhost/admin).
- pentru user normal:
  - vizualizare listă cu produsele disponibile;
  - vizualizarea ierarhie de categorii a produselor;
  - filtrarea produselor prin selectarea unei categorii.
- pentru administrator:
  - toate operațile unui user normal;
  - adăugare produse într-o categorie;
  - stergere produse;
  - adăugarea de subcategorii categoriilor existente;
  - ștergerea de subcategorii.

1.3 Cerinte non-functionale

Urmatoarele cerinte trebuie respectate:

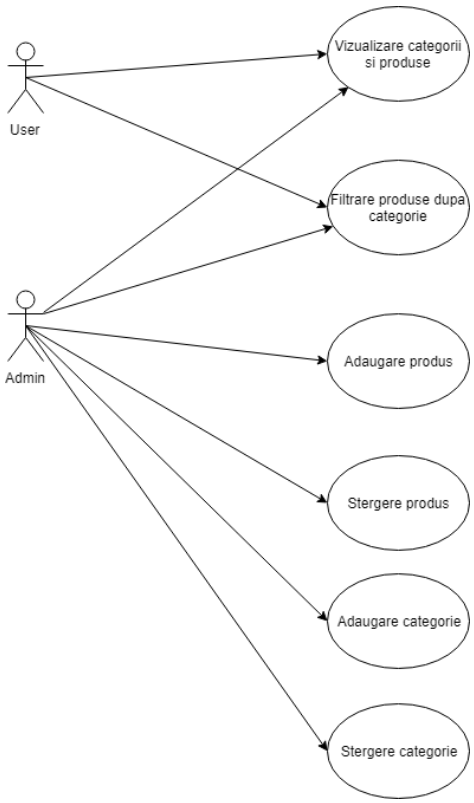
- utilizarea unui framework ce implementează patternul MVC (Spring Boot);
- utilizarea unui frameword Object-Relational Mapping (JPA);
- datele stocate într-o bază de date (MySql);
- utilizaera pattern-ului Composite;

- timpul de raspuns al sistemului pentru orice operatie trebuie sa fine mai mic de 2 secunde;
- arhitectura sistemului trebuie sa permina adaugarea de noi operatii sau modificarea celor curente;

2 Use Case Model

Use case: Adăugarea unui produs;
Level: User-goal;
Actor princital: Admin;
Scenariu de succes: După analizarea ierarhiei, utilizatorul introduce numele, prețul și id-ul (categoriei în care se dorește adăugarea) și se apasă butonul ADD. Produsul va apărea apoi în lista de produse;
Extensii: Operatia poate eșua dacă:

- id-ul introdus nu corespunde nici unei categorii;
- în cămpul pentru preț se introduc date eronate;
- intervine o problemă în baza de date.



3 Design-ul arhitectural al sistemului

3.1 Descrierea modelului arhitectural

Este utilizat pattern-ul arhitectural MVC.
Aplicatia este compusa din trei nivele:

- View - contine interfata cu utilizatorul, în cazul nostru este web;
- Controller - leaga Modelul de View și procesează toate cererile;
- Model - oferă acces structurat la date. Acesta este împărtit în alte trei părți:
  - Entities - structurile de date;
  - Repositories - accesul la baza de date;
  - Services - combină Entities și Repositories pentru a returna date structurate.

3.2 Diagrame

Urmatoarea diagrama prezinta structura pachetelor din sistem :

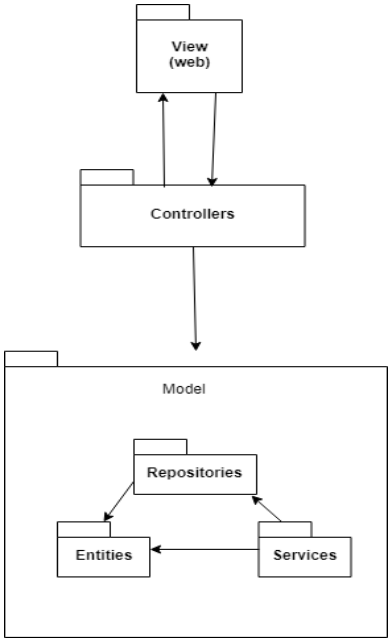


Diagrama de componente:

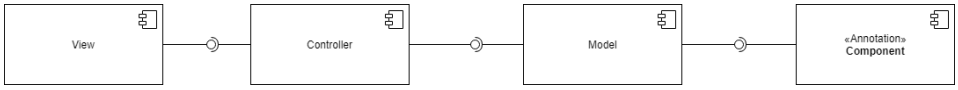
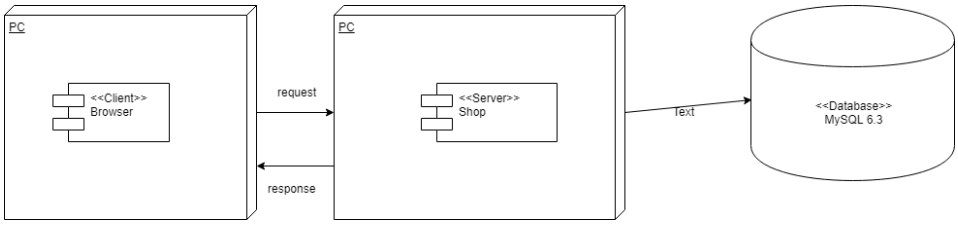
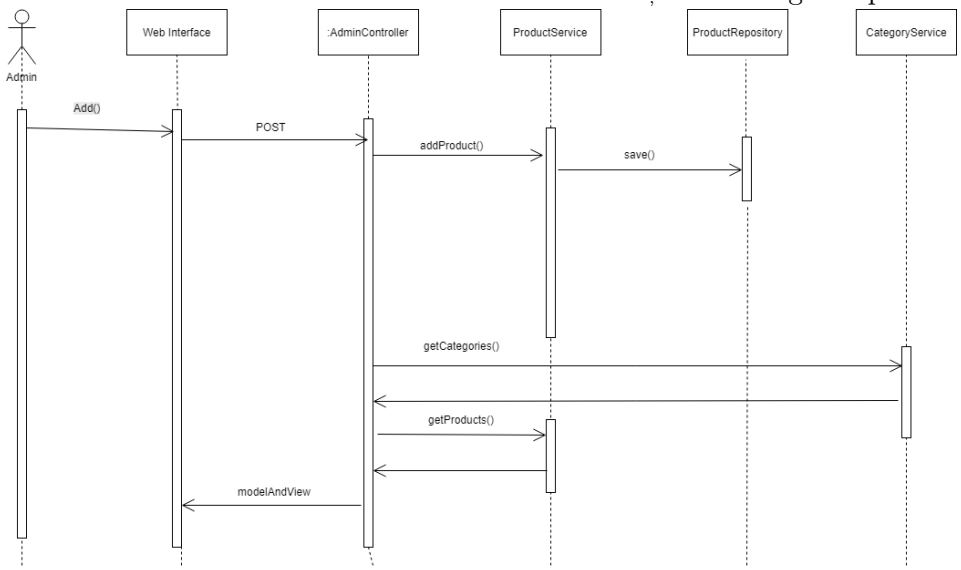


Diagrama de deployment:



4 Diagrame de secventa UML

Consideram scenariul în care administratorul dorește să adauge un produs:



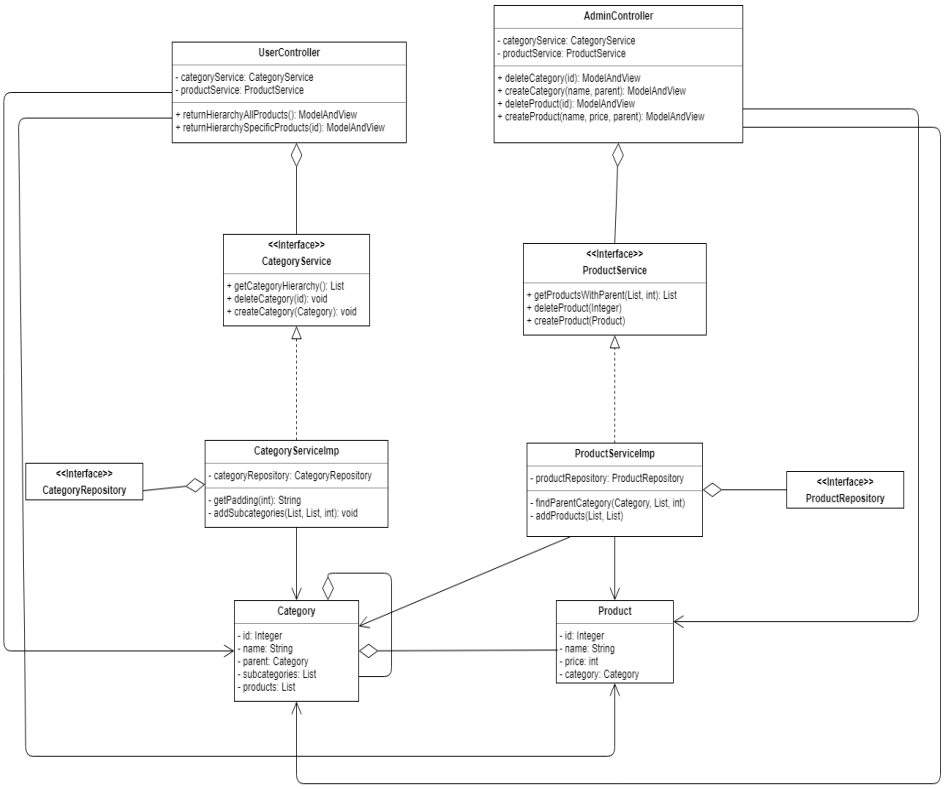
5 Design-ul claselor

5.1 Descriere design pattern-uri utilizate

Au fost utilizate urmatoarele:

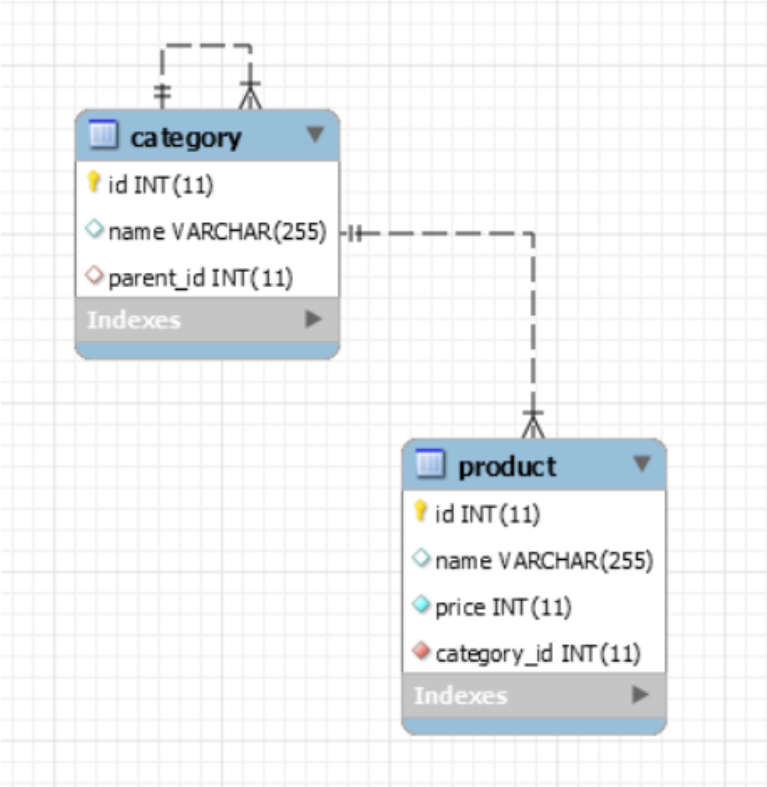
- Composite - Clasele Product și Category formează o ierarhie, o instanță a clasei Category conține o listă de instanțe a clasei Product;
- Repository - ProductRepository și CategoryRepository se ocupă de accesul la baza de date;

5.2 Diagrama de clase



6 Model de date

Baza de date in care sunt stocate informatiile este urmatoare:



Dupa cum se poate, vedea exista două tabele:

- Product - atributul category_id face legătura unui produc cu o categorie;
- Category - atributul parent_id leaga o categorie de o subcategorie.

7 Testarea sistemului

S-a folosit o strategie de testare incrementală, funcționalitățile au fost verificate pe măsură ce au fost adăugate in sistem. Astfel, principalele metode de testare au fost data-flow (au fost date anumite intrări si s-a observat raspunsul sistemului), white-box (codul a fost disponibil).

8 Bibliografie

Pentru utilizare Spring Boot și JPA:

- `https://www.youtube.com/watch?v=msXL2oDexqw&list=PLqq-6Pq4lTTbx8p2oCgcAQGQyqN8XeA1x`

Pentru anumite erori și html:

- `https://stackoverflow.com/`
- `https://www.youtube.com/`