***Sistem de management al comenzilor***

# Obiective

Creearea unei aplicatii menite sa permita simularea unui sistem bazat pe clienti, produse si comenzi, in vederea simularii unui magazin, menit sa proceseze comenzile anumitor produse din stoc primite de catre un anumit client prezent in data de baze si implementarea de algoritmi multiplii cu scopul de a adauga cat mai rapid si mai optim in functie de baza de date existenta. Astfel, se identifica urmatoarele obiective secundare in vederea obtinerii unei aplicatii viabile si sustenabile:

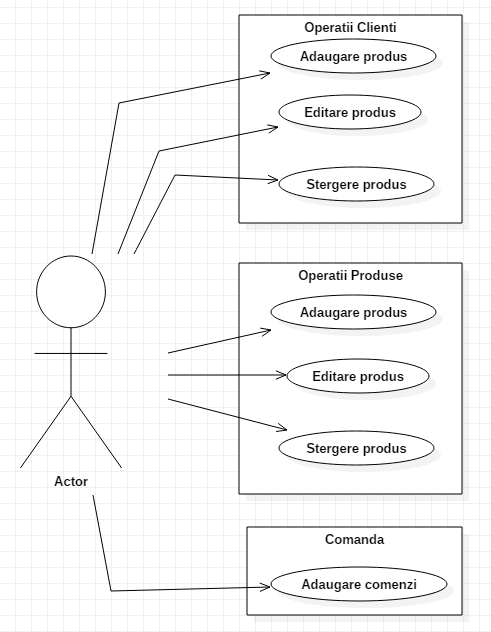
|  |  |
| --- | --- |
| Obiective |  |
| Analiza cerintelor functionale  (Capitolul 2) | * Identificarea scenariilor de functionare din punct de vedere al utilizatorului si valorile de intrare introduse de acesta * Identificarea cerintelor functionale: operatiile pe date ce trebuie efectuate de catre sistem * Modelarea procesului de functionare al aplicatiei |
| Proiectarea  (Capitolul 3) | * Structurarea proiectului: impartirea sa in clase si pachete, diagrama UML * Definirea unor structuri de date ce sa faciliteze implementare precum si lizibilitatea codului * Analiza algoritmilor ce trebuie implementati in vederea cresterii performantei |
| Implementarea propriu-zisa (Capitolul 4) | * Punerea in practica a pasilor anterior * Urmarea pasilor de proiectare in mod sistemic pentru evitarea scrierii de cod redundant * Scrierea de cod lisibil si usor de interpretat prin limitarea liniilor de cod (maxim 30 de linii per metoda, respectiv maxim 300 de linii per clasa) |
| Analiza rezultatelor (Capitolul 5) | * Conceperea de teste menite sa verifice fiabilitatea si corectitudinea implementarii tuturor cerintelor funtionale * Interpretarea rezultatelor si descrierea eventualelor erori aparute sau identificate prin acestea |
| Dezvoltare  (Capitolul 6) | * Analiza retrospectiva a proiectului surprinzand dificultatile intalnite * Prezentarea posibilitatilor de dezvoltare ulterioare ale aplicatiei |

# Analiza problemei, modelare, scenarii, cazuri de utilizare

Functionalitatile identificate in urma analizei cerintelor stipulate de catre client sunt:

1. Conexiunea aplicatiei la baza de date

* Creearea unei baze de date stabile si sustenabile
* Utilizarea sintaxei corecte de SQL
* Creearea a minim 3 tabele reprezentate de client, produs si comanda
* Respectarea paradigmelor de utlizarea a bazelor de date

Diagrama use-case: 

In cele ce urmeaza vor fi prezentate cazurile de utilizare ale aplicatiei:

*Actor principal: Utilizator al aplicatiei ( Manager magazin etc.)*

**Use case adaugare client:**

1. Actorul deschide aplicatia
2. Actorul alege din fereastra client optiunea Add Client
3. Actorul introduce datele necasare
4. Actorul apasa pe butonul „Send”
5. Aplicatia incepe sa proceseze datele
6. Aplicatia verifica daca datele respecta formatul cerut
7. Aplicatia creeaza noul client
8. Aplicatia introduce in baza de date noul client
9. Aplicatia afiseaza in id-ul cu care a fost introdus noul client
10. Aplicatia actualizeaza tabelul de clienti
11. Actorul alege sa introduca alt client sau sa se intoarca la meniul principal

**Use case adaugare produs:**

1. Actorul deschide aplicatia
2. Actorul alege din fereastra client optiunea Add Product
3. Actorul introduce datele necasare
4. Actorul apasa pe butonul „Send”
5. Aplicatia incepe sa proceseze datele
6. Aplicatia verifica daca datele respecta formatul cerut
7. Aplicatia creeaza noul produs
8. Aplicatia introduce in baza de date noul produs
9. Aplicatia afiseaza in id-ul cu care a fost introdus noul produs
10. Aplicatia actualizeaza tabelul de produse
11. Actorul alege sa introduca alt produs sau sa se intoarca la meniul principal

**Use case adaugare comanda:**

1. Actorul deschide aplicatia
2. Actorul introduce datele necasare
3. Actorul apasa pe butonul „Send”
4. Aplicatia incepe sa proceseze datele
5. Aplicatia verifica daca datele respecta formatul cerut
6. Aplicatia creeaza noua comanda
7. Aplicatia introduce in baza de date noua comanda
8. Aplicatia afiseaza in id-ul cu care a fost introdusa noua comanda
9. Actorul alege sa introduca alta comanda

**Use case stergere client:**

1. Actorul deschide aplicatia
2. Actorul alege din fereastra client optiunea Delete Client
3. Actorul introduce datele necasare
4. Actorul apasa pe butonul „Send”
5. Aplicatia incepe sa proceseze datele
6. Aplicatia verifica daca datele respecta formatul cerut
7. Aplicatia sterge din baza de date clientul
8. Aplicatia afiseaza ca stergerea s-a realizat cu succes
9. Aplicatia actualizeaza tabelul de clienti
10. Actorul alege sa stearga alt client sau sa se intoarca la meniul principal

**Use case stergere produs:**

1. Actorul deschide aplicatia
2. Actorul alege din fereastra client optiunea Delete Product
3. Actorul introduce datele necasare
4. Actorul apasa pe butonul „Send”
5. Aplicatia incepe sa proceseze datele
6. Aplicatia verifica daca datele respecta formatul cerut
7. Aplicatia sterge din baza de date produsul
8. Aplicatia afiseaza ca stergerea s-a realizat cu succes
9. Aplicatia actualizeaza tabelul de produse
10. Actorul alege sa stearga alt produs sau sa se intoarca la meniul principal

**Use case actualizeaza client:**

1. Actorul deschide aplicatia
2. Actorul alege din fereastra client optiunea Edit Client
3. Actorul introduce datele necasare
4. Actorul apasa pe butonul „Send”
5. Aplicatia incepe sa proceseze datele
6. Aplicatia verifica daca datele respecta formatul cerut
7. Aplicatia actualizeaza clientul in baza de date
8. Aplicatia afiseaza daca actualizarea a avut succes
9. Aplicatia actualizeaza tabelul de clienti
10. Actorul alege sa actualizeze alt client sau sa se intoarca la meniul principal

**Use case actualizeaza produs:**

1. Actorul deschide aplicatia
2. Actorul alege din fereastra client optiunea Edit Product
3. Actorul introduce datele necasare
4. Actorul apasa pe butonul „Send”
5. Aplicatia incepe sa proceseze datele
6. Aplicatia verifica daca datele respecta formatul cerut
7. Aplicatia actualizeaza produsul in baza de date
8. Aplicatia afiseaza daca actualizarea a avut succes
9. Aplicatia actualizeaza tabelul de produse
10. Actorul alege sa actualizeze alt produs sau sa se intoarca la meniul principal

**Use case-uri alternative:**

1. Datele introduse nu respecta formatul

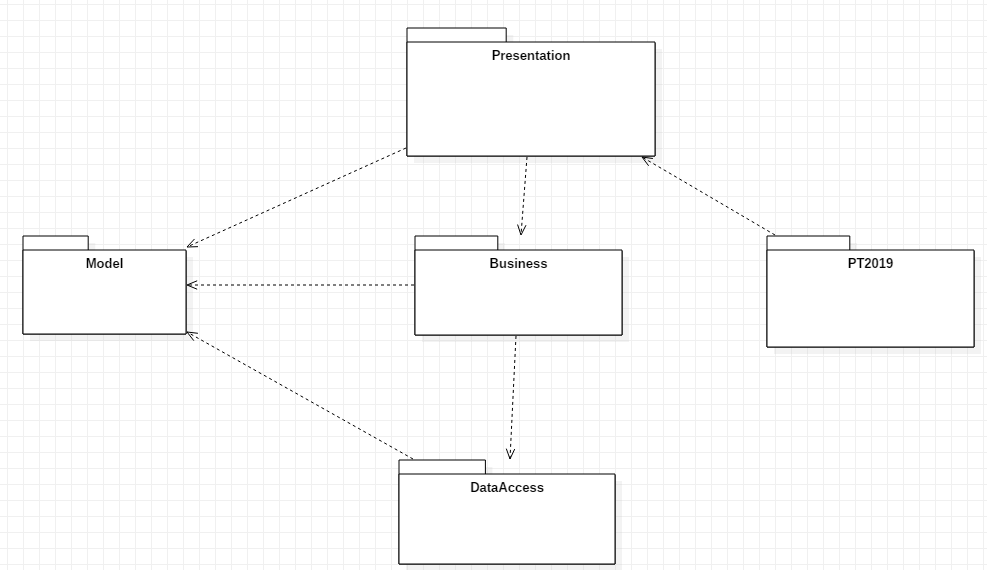
* Aplicatia anunta actorul cu mesajul „input invalid”
* Se revine la pasul 3

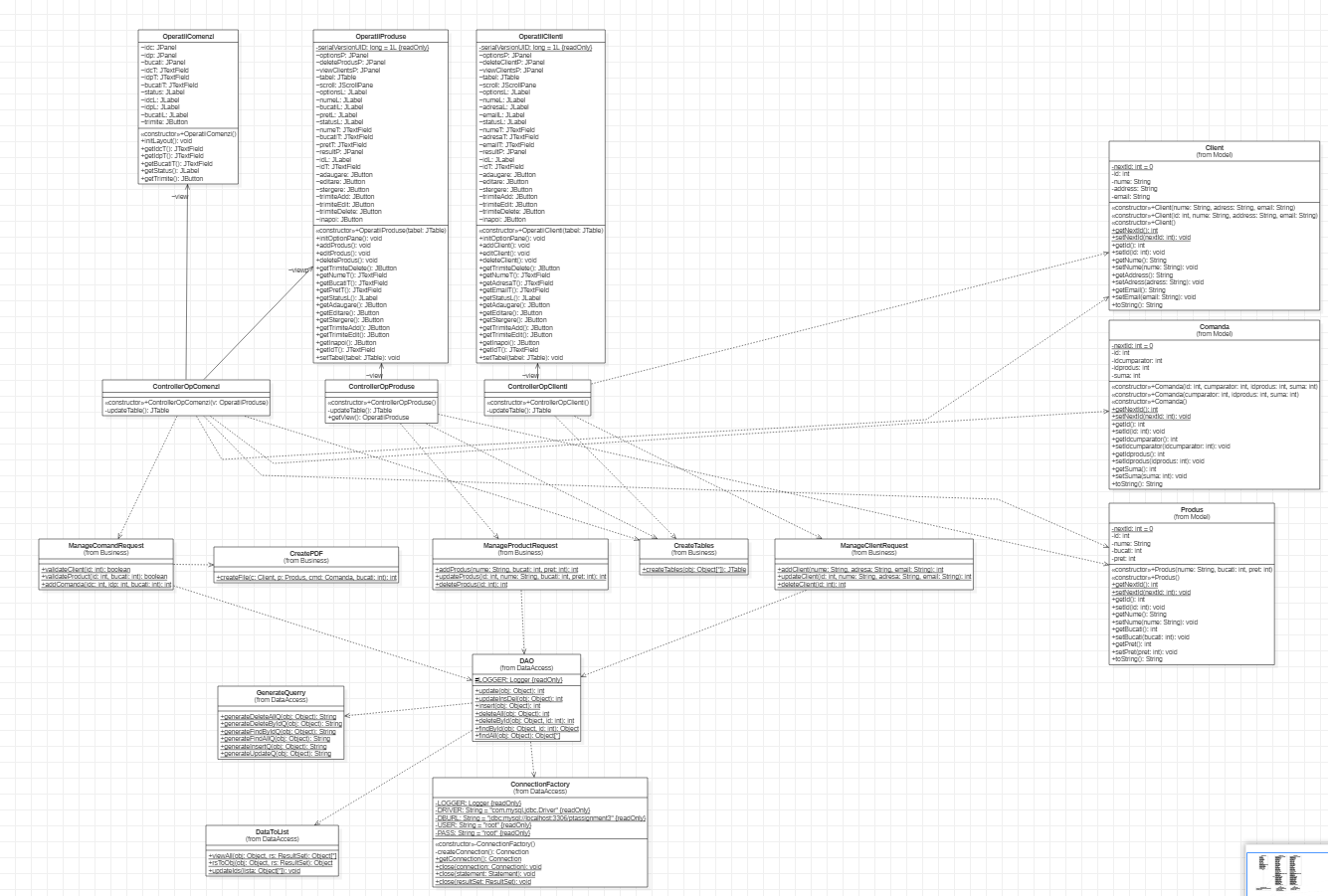
1. Datele introduse sunt invalide

* Aplicatia anunta actorul cu mesajul „input invalid”
* Se revine la pasul 3

1. Datele introduse nu corespund cu cele din tabel

* Aplicatia anunta actorul cu mesajul „id invalid”
* Se revine la pasul 3





# 3.Implementare

**Pachetul Model**

A. Clasa Client

Contine Campurile:

private static int *nextId*=0;

private int id;

private String nume;

private String address;

private String email;

B. Clasa Produs

Contine Campurile:

private static int *nextId*=0;

private int id;

private String nume;

private int bucati;

private int pret;

C. Clasa Comanda

Contine Campurile:

private static int *nextId*=0;

private int id;

private int idcumparator;

private int idprodus;

private int suma;