# **DOCUMENTATIE**

# TEMA 3

NUME STUDENT: Şulea Alexandra-Maria GRUPA: 30222

# **CUPRINS**

1.	Obiectivul temei	3
	Analiza problemei, modelare, scenarii, cazuri de utilizare	
	Proiectare	
	Implementare	
	Rezultate	
	Concluzii	
	Bibliografie	

#### 1. Obiectivul temei

**Obiectivul principal:** Dezvoltarea unei aplicații pentru gestionarea comenzilor și a produselor într-un magazin online.

Obiective secundare: Implementarea interfetei de utilizator, dezvoltarea functionalitatilor de adaugare, modificare, stergere a comenzior si produselor, asigurarea actualizarii stocului de produse si functia de comenzile plasate, asigurarea actualizarii stocului de produse in functie de comenzile plasate, verificarea corectitudinii functionarii aplicatiei si asigurarea absentei erorilor.

### 2. Analiza problemei, modelare, scenarii, cazuri de utilizare

### Cadrul de cerințe funcționale:

- Crearea, modificarea şi ştergerea de clienţi şi produse.
- Plasarea, modificarea și ștergerea de comenzi.
- Actualizarea stocului de produse în funcție de comenzile plasate.

#### Cazuri de utilizare:

### 1. Adăugare client/produs:

- Utilizatorul introduce datele clientului/produsului.
- Se validează datele.
- Datele sunt adăugate în baza de date.

#### 2. Plasare comandă:

- Utilizatorul selectează produsul și cantitatea dorită.
- Se verifică disponibilitatea în stoc.
- Se actualizează stocul.
- Se adaugă comanda în baza de date.

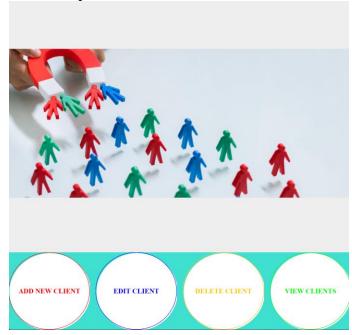
# 3. Proiectare



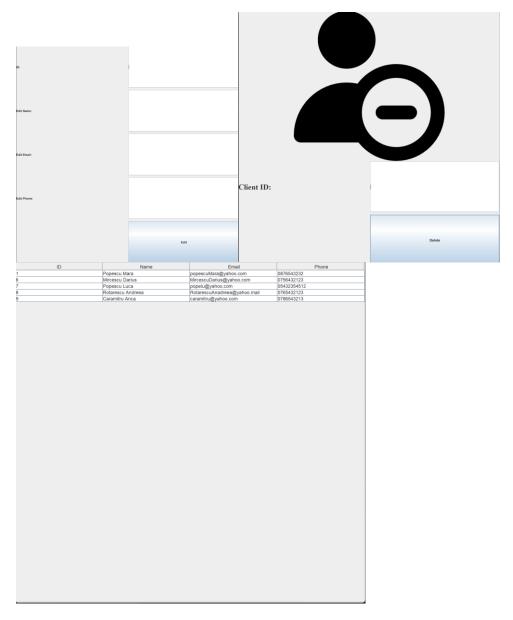
# Interfata principal



# Interfetele pentru client



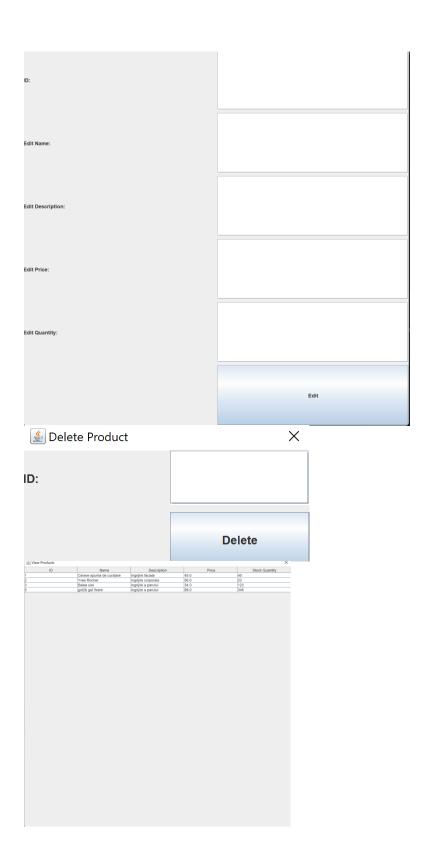


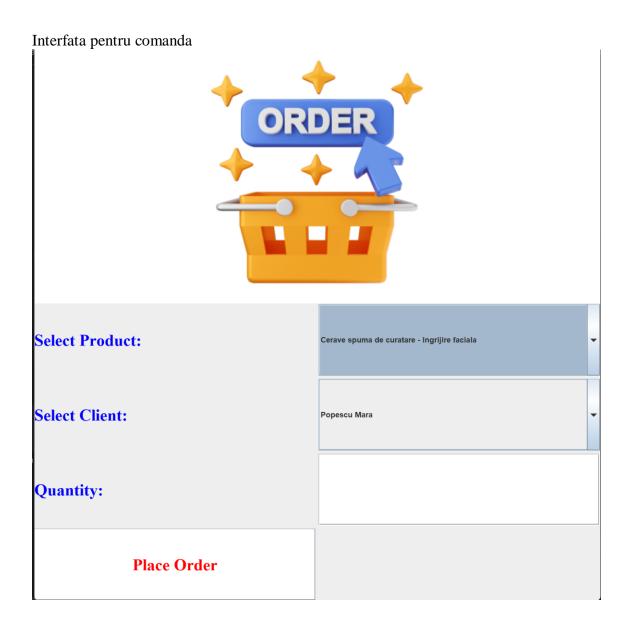


Interfetele pentru produs



Name:			
Description:			
Price:			
Quantity:			
		Add	





# 4. Implementare

Pachetul Validators:

ClientEmail: clasa care verifica email-ul;

ClientName: clasa care verifica numele;

ClientPhone: clasa care verifica numarul de teelfon;

Pachetul Connection:

ConnectionFactory: se ocupa cu conexiunea la baza de date;

#### Pachetul Dao:

ClientDao: se ocupa cu operatiile de adaugare, stergere, editare si vizualizare a datelor din baza de date pentru client;

OrderDao: se ocupa cu adaugarea si facturarea unei comenzi in baza de date;

ProductDao: se ocupa cu adaugarea, stergerea, editarea si vizualizarea datelor din baza de date pentru produs;

Pachetul Model:

Bill: reprezintă o entitate care definește o factură asociată unei comenzi.

Client: reprezintă o entitate care definește un client în cadrul sistemului. Product: reprezintă un produs disponibil pentru vânzare în cadrul sistemului;

Order: reprezintă o comandă plasată de un client pentru achiziționarea unui produs.

ClientFrame: interfetele pentru categoria "client"; OrderFrame: interfetele pentru categoria" order"; ProductFrame: interfetele pentru "product";

Start: clasa pentru pornirea aplicatiei;

#### 5. Rezultate

Se vor prezenta scenariile pentru testare. In cazul in care in cerinta temei se specifica sa se faca testare unitara cu utilitarul JUnit, se vor integra in acesta sectiune rezultate testari cu JUnit.

#### 6. Concluzii

Se vor prezenta concluziile, ce s-a invatat din tema, posibile de dezvoltari ulterioare.

### 7. Bibliografie

1. Java Swing Tutorial - Oracle Documentation

2. Java Documentation - Oracle

3.https://www.youtube.com/watch?v=FEb2Pt9ymes&pp=ygUgY3VtIGZhY2kgYnV0b2FuZSByb3R 1bmRIIGIuIGphdmE%3D