DOCUMENTATIE

TEMA *3*

NUME STUDENT: …Avram Darius

GRUPA: …………5………

# CUPRINS

[1. Obiectivul temei 3](file:///C:\foldere\Darius\PT2023_5_Avram_Darius_Assignment_1\PT2023_Documentatie_RO.doc#_Toc95297885)

[2. Proiectare 3](file:///C:\foldere\Darius\PT2023_5_Avram_Darius_Assignment_1\PT2023_Documentatie_RO.doc#_Toc95297887)

[3. Implementare 4](file:///C:\foldere\Darius\PT2023_5_Avram_Darius_Assignment_1\PT2023_Documentatie_RO.doc#_Toc95297888)

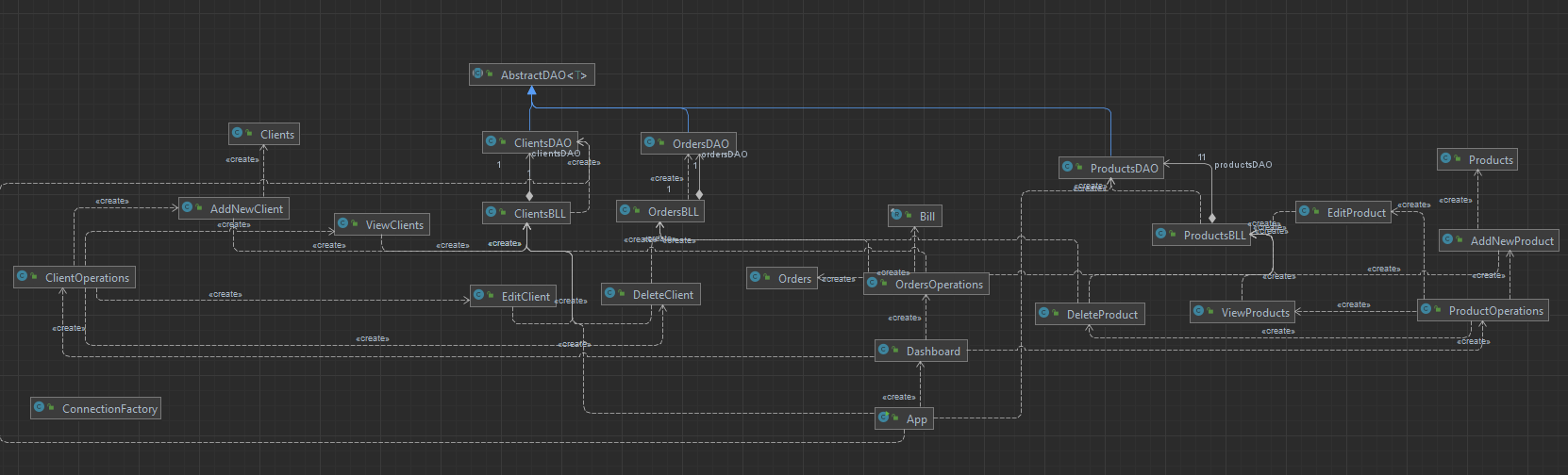
[4. Concluzii 4](file:///C:\foldere\Darius\PT2023_5_Avram_Darius_Assignment_1\PT2023_Documentatie_RO.doc#_Toc95297890)

[5. Bibliografie 5](file:///C:\foldere\Darius\PT2023_5_Avram_Darius_Assignment_1\PT2023_Documentatie_RO.doc#_Toc95297891)

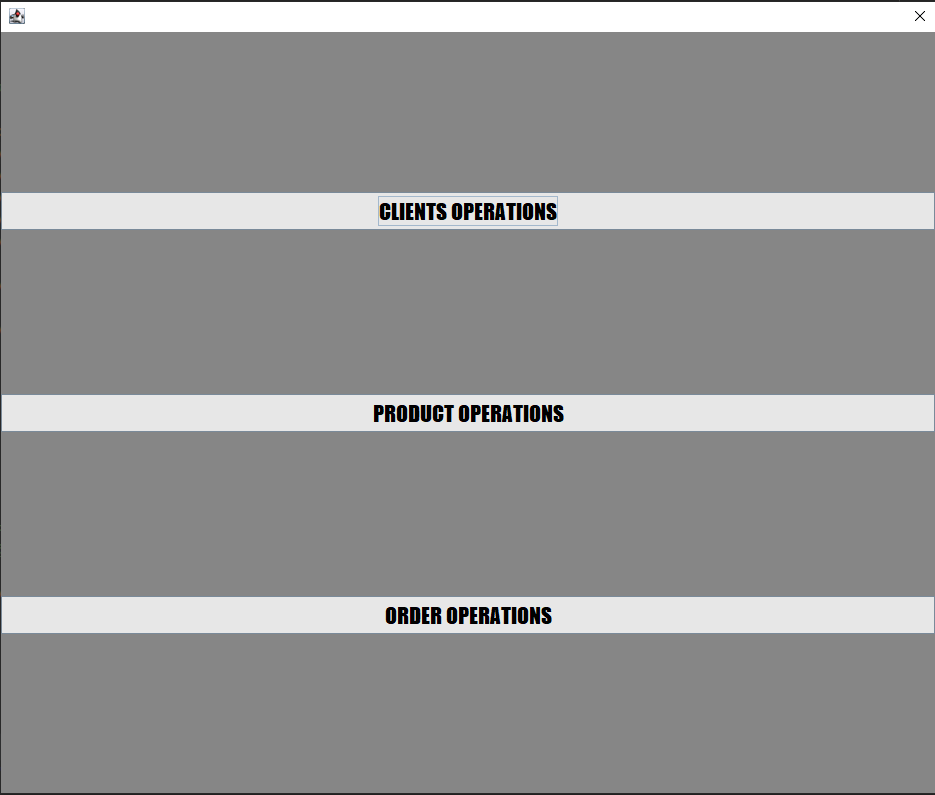
# Obiectivul temei

Obiectivul acestei teme este de a dezvolta o aplicație pentru un depozit, folosind baze de date relaționale pentru a stoca datele despre produse, clienți și comenzi. Aplicația ar trebui să urmeze un model de arhitectură stratificat, cu clase de model reprezentând modele de date, clase de business logic care gestionează logica aplicației, clase de prezentare pentru GUI și clase de acces la date pentru interacțiunea bazei de date.

# Proiectare



# 



# Implementare

* **bll**

-clasa ClientsBLL este responsabilă de operațiile de logica de afaceri pentru manipularea datelor despre clienți. Aceasta interacționează cu clasa ClientsDAO pentru accesul la date și cu clasele GUI pentru interacțiunea cu interfața utilizatorului.

* **connection**

-clasa ConnectionFactory este responsabilă de conectarea la baza de date

* **dao**

-clasa AbstractDAO este o clasă abstractă generică care furnizează funcționalitate comună pentru operațiile de acces la date (select, insert, update, delete) pentru o entitate specifică. Această clasă este extinsă de clasele DAO specifice (ClientsDAO, OrdersDAO, ProductsDAO) pentru a implementa operațiile specifice entității respective.

* **gui**

-clasele din acest pachet sunt responsabile pentru prezentarea și interacțiunea cu utilizatorul în cadrul aplicației.

* **model**

-clasa Clients reprezintă un model de date pentru un client, metode de getter și setter

# Concluzii

Prin realizarea acestei teme, am avut oportunitatea de a învăța și de a pune în practică câteva concepte și principii importante în dezvoltarea aplicațiilor. Am înțeles importanța împărțirii aplicației în pachete, cum ar fi model, bll, dao și gui, pentru a obține un design modular și ușor de gestionat. Am avut ocazia să lucrez cu baze de date relaționale, să înțeleg structura și normalizarea datelor, precum și să utilizez clasele dao pentru a interacționa cu baza de date.

Nu în ultimul rând, am dezvoltat clasele din pachetul gui, pentru a oferi o interfață prietenoasă utilizatorilor, permițându-le să interacționeze eficient cu aplicația.

În concluzie, această temă mi-a oferit o perspectivă mai amplă asupra dezvoltării aplicațiilor și mi-a permis să aplic cunoștințele și conceptele într-un proiect interesant.

# Bibliografie

<https://stackoverflow.com/questions/tagged/java>

<https://www.geeksforgeeks.org/java>   
<https://www.javatpoint.com/java-tutorial>  
<https://www.javacodegeeks.com>