

Documentatie

Tema 3

Nume: Saca Victor-Valentin

1. Obiectivul temei

Obiectivul acestui proiect este de a dezvolta o aplicație Java bazată pe o interfață grafică (GUI) pentru gestionarea clienților, produselor și comenzilor într-un sistem de tip CRUD (Create, Read, Update, Delete).

Utilizatorii vor putea să adauge, să vizualizeze, să actualizeze și să șteargă informații despre clienți și produse, precum și să gestioneze comenzile, inclusiv verificarea stocului disponibil.

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

Analiza problemei

Într-un sistem de gestionare a comenzilor și a stocurilor, utilizatorii trebuie să poată:

Adăuga și vizualiza informații despre clienți.

Adăuga și vizualiza informații despre produse.

Crea și vizualiza comenzi.

Asigura că stocul este suficient înainte de a crea o comandă.

Actualiza și șterge informațiile existente.

Modelare

Modelele principale implicate sunt:

Client: conține informații despre clienți (ID, nume, email).

Product: conține informații despre produse (ID, nume, preț, stoc).

Order: conține informații despre comenzi (ID, ID client, ID produs, cantitate, dată).

Scenarii și cazuri de utilizare

UC1: Adăugare client: Utilizatorul introduce numele și email-ul unui nou client.

UC2: Vizualizare clienți: Utilizatorul vizualizează lista clienților existenți.

UC3: Actualizare client: Utilizatorul modifică informațiile unui client existent.

UC4: Ștergere client: Utilizatorul șterge un client existent.

UC5: Adăugare produs: Utilizatorul introduce informațiile unui nou produs.

UC6: Vizualizare produse: Utilizatorul vizualizează lista produselor existente.

UC7: Actualizare produs: Utilizatorul modifică informațiile unui produs existent.

UC8: Ștergere produs: Utilizatorul șterge un produs existent.

UC9: Creare comandă: Utilizatorul selectează un client, un produs și introduce cantitatea dorită pentru a crea o comandă.

UC10: Verificare stoc: Sistemul verifică stocul disponibil înainte de a finaliza comanda.

3. Proiectare

Aplicația este proiectată folosind Java și JDBC pentru conectarea la baza de date. Interfața grafică este construită folosind Swing.

Arhitectura aplicației

DAO (Data Access Object): Pentru fiecare entitate (Client, Product, Order) există un DAO corespunzător care gestionează operațiile CRUD.

Model: Entitățile care reprezintă datele (Client, Product, Order).

GUI: Interfața grafică pentru interacțiunea utilizatorului cu sistemul.

4. Implementare

org.model.Client - Reprezintă entitatea Client, conținând attributele client_id, name și email, împreună cu constructorii necesari și metodele getter și setter.

org.model.Product - Reprezintă entitatea Product, având attributele product_id, name, price și stock, împreună cu constructorii necesari și metodele getter și setter.

org.model.Orders - Reprezintă entitatea Orders, cu attributele order_id, client_id, product_id, quantity și order_date, incluzând constructorii și metodele getter și setter pentru gestionarea comenzilor.

org.dao.AbstractDAO<T> - Clasă abstractă generică ce furnizează funcționalități de bază pentru operațiile CRUD (Create, Read, Update, Delete) asupra entităților din baza de date. Utilizează reflecția pentru a maparea obiectelor Java la înregistrările din baza de date.

org.dao.ClientDAO - Extinde AbstractDAO pentru a gestiona operațiile specifice entității Client, cum ar fi

inserarea, actualizarea și ștergerea clienților din baza de date.

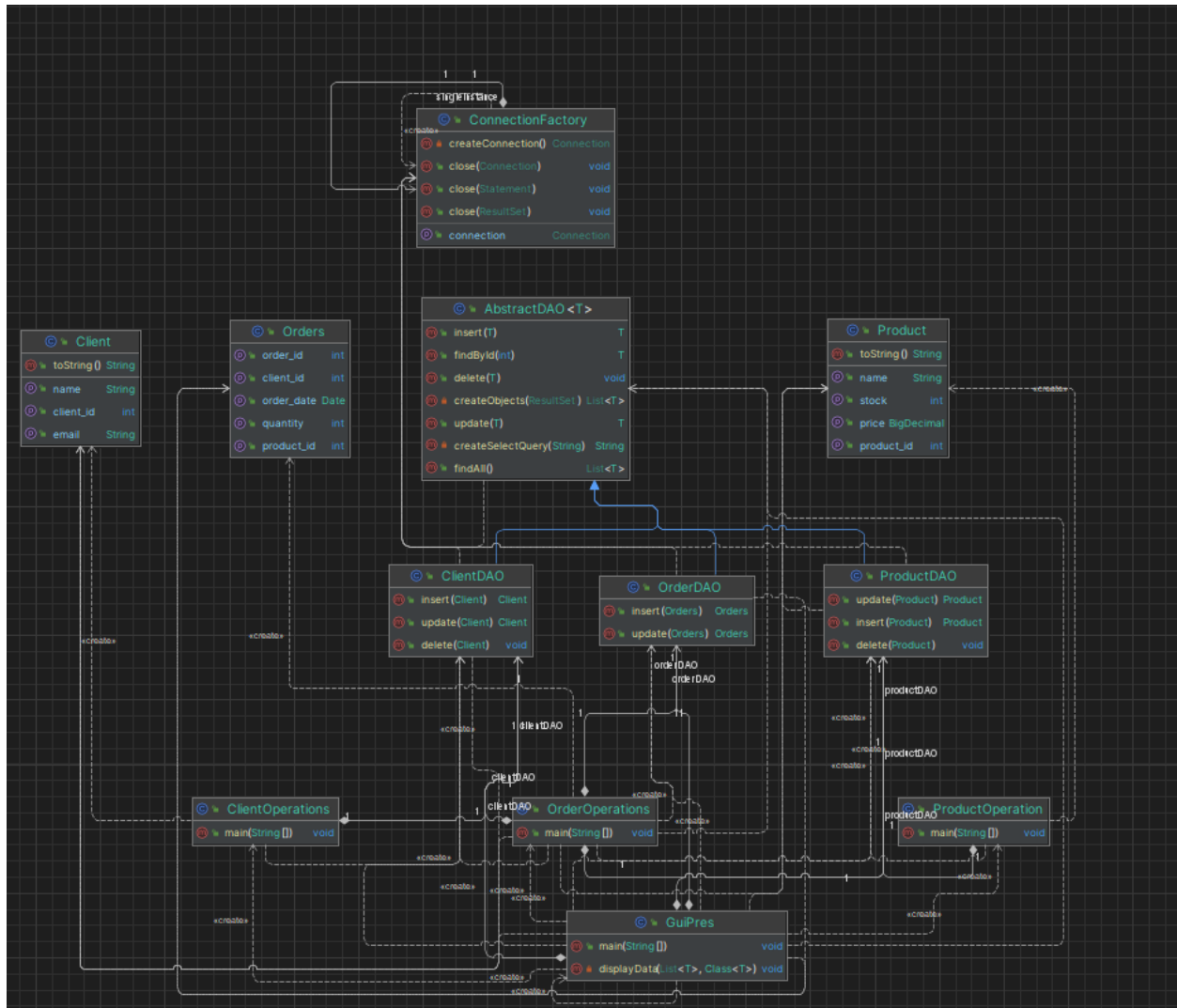
org.dao.ProductDAO - Extinde AbstractDAO pentru a gestiona operațiile specifice entității Product, inclusiv inserarea, actualizarea și ștergerea produselor din baza de date.

org.dao.OrderDAO - Extinde AbstractDAO pentru a gestiona operațiile specifice entității Orders, permitând inserarea, actualizarea și ștergerea comenzilor din baza de date.

org.presentation.ClientOperations - Interfață grafică pentru gestionarea operațiunilor legate de clienți, permițând utilizatorilor să adauge, actualizeze și șteargă clienți printr-o interfață prietenoasă.

org.presentation.ProductOperations - Interfață grafică pentru gestionarea operațiunilor legate de produse, permițând utilizatorilor să adauge, actualizeze și șteargă produse printr-o interfață intuitivă.

org.presentation.OrderOperations - Interfață grafică pentru crearea și gestionarea comenzilor de produse, permitând utilizatorilor să selecteze produse și clienți existenți, să introducă cantitatea dorită și să plaseze comenzi, gestionând stocurile de produse.



6. Bibliografie

- <https://stackoverflow.com/>

- <https://www.jetbrains.com/>

- <https://dsrl.eu/courses/pt/> (resursele cu fisiere pdf ale facultatii)