**DOCUMENTATIE**

**TEMA 3**

LASZLO BOGDAN GHEORGHE

GRUPA: 30222

Cuprins

[1. Obiectivul temei 3](#_Toc134979541)

[2. Analiza problemei, modelare, scenarii, cazuri de utilizare 3](#_Toc134979542)

[3. Proiectare 3](#_Toc134979543)

[4. Implementare 4](#_Toc134979544)

[5. Rezultate 5](#_Toc134979545)

[6. Concluzii 11](#_Toc134979546)

[7. Bibliografie 11](#_Toc134979547)

# Obiectivul temei

Obiectivul principal al temei este de a simula un manager de comenzi.

Obiective secundare:

* Folosirea Reflection-ului pentru a implementa accesul la baza de date
* Crearea unui JavaDoc
* Crearea unui backup pentru baza de date folosind MySQL Dump

# Analiza problemei, modelare, scenarii, cazuri de utilizare

Va trebui sa implementam o interfata pentru manageriat de comenzi. Fiecare buton va ajea cate o functionalitate sub-intelesa.

Fiecare buton de pe pagina principala duce la o alta pagina care are tabelul cu informatiile privitoare la ce dorim(Client,Produs,Comanda etc.) si cateva butoane, in functie de tabelul ales.

In pagina de client si cea de produs exista 3 butoane, add, care adauga un obiect nou cu informatiile intruduse, update, care modifica un obiect cu valoriile introduse cu id-ul introdus si delete, care sterge un obiect in functie de id-ul introdus.

In pagina de Comanda este doar un buton, order, care, apasandu-l ne trimite la o pagina unde selectam clientul ce doreste sa comande, produsul si cantitatea ce doreste sa o comande. Daca cantitatea este mai multa decat cea care este in stoc va aparea un mesaj de eroare, daca totul este ok, se va adauga un obiect nou de comanda cu id-ul sau, id-ul clientului, id-ul produsului cantitatea comandata si totalul platit.

In pagina Bill se adauga un obiect atunci cand se face o comanda, adauganu-se automat un id si un total al comenzii.

# Proiectare

Proiectul este impartit in clasele:

* Client
* Product
* This\_Order
* Bill
* ClientDAO
* ProductDAO
* This\_OrderDAO
* BillDAO
* AbstractDAO
* ConnectionFactory
* View
* A screenshot of a computer

  Description automatically generated with medium confidenceMain

# Implementare

In clasa AbstractDAO am implementat metodele:

* createSelectAllQuery
* createInsertQuery
* createUpdateQuery
* createDeleteQuery
* createSelectQuery

care creaza query-urile pentru baza de date mysql.

Am mai implementat metodele FillTable, care populeaza o tabela de tipul JTable, findAll, care returneaza toate obiectele dintr-un table, insert, care insereaza un obiect nou intr-o tabela, findById, care returneaza un obiect dupa id-ul acestuia din tabela, update, care modifica valorile unui obiect dintr-o tabela in functie de id-ul acestuia si delete, care sterge un obiect in functie de id-ul acestuia din tabela.

In clasa Product, Client, ,This\_Order, Bill am implementat 3 constructori, settere si gettere si o metoda toString.

In clasa View am implementat constructorul View care se ocupa de toata interfata proiectului.

In clasa Main se creeaza un nou View.

# Rezultate

A screenshot of a computer

Description automatically generated with medium confidence

Pagina principala.

A screenshot of a computer

Description automatically generated

La apasarea butonului clients se deschide fereastra de client.

A screenshot of a computer

Description automatically generated

La apasarea butonului Products se deschide fereastra de produse.

A screenshot of a computer

Description automatically generated with medium confidence

La apasarea butonului Orders se deschide fereastra de comenzi.

A screenshot of a computer

Description automatically generated with medium confidence

La apasarea butonului Bill se deschide fereastra de bonuri.

A screenshot of a computer

Description automatically generated with medium confidence

La apasarea butonului Add din fereastra CLIENTS se deschide o fereastra in care introducem datele clientului care dorim sa il inseram.

A screenshot of a computer

Description automatically generated with medium confidence

La apasarea buronului Update se deschide fereastra da modificare a unui client, completandu-se id-ul clientului care se doreste a fi modifcat si datele ce vor fi modificate.

A screenshot of a computer

Description automatically generated with medium confidence

La apasarea butonului Delete de pe pagina de clienti se deschide op noua fereastra de stergere a unui client, completandu-se id-ul clientului ce se doreste a fi sters.

Pe pagina de produse sunt aceleasi operatii si ferestre.

A screenshot of a computer

Description automatically generated with medium confidence

La apasarea butonului Order de pe pagina de comenzi se deschide o noua pagina in care este selectat clientul dorit din lista de clienti precum si produsul dorit si se completeaza cantitatea ce este dorita sa se comande. Daca cantitatea dorita este mai mare decat cantitatea din stock atunci va aparea o fereastra de eroare si nu se va efectua, iar daca totul este ok, se va efectua.

A screenshot of a computer

Description automatically generated with medium confidence

# Concluzii

Am invatat cum se lucrez cu Reflection, am invatat cum se creaza un JavaDoc si am invatat cum se populeaza un JTable.

# Bibliografie

<https://dsrl.eu/courses/pt/>