**OrderManagement**

Implementati în considerare o aplicație OrderManagement pentru procesarea comenzilor clienților pentru un depozit. Bazele de date relaționale sunt utilizate pentru a stoca produsele, clienții și comenzile. În plus, aplicația ar trebui să fie structurată în pachete folosind o arhitectură stratificată și ar trebui să utilizeze (minim) următoarele clase:

• Model classes - the data models of the application

• Business Logic classes – implement the application logic

• Presentation classes – implement the user input/output

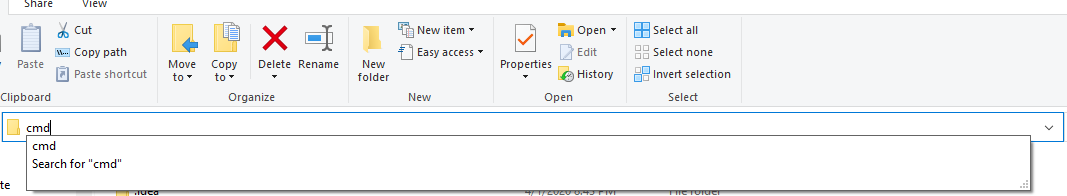
• Data access classes - implement the access to the database

Aplicația ar trebui să permită procesarea comenzilor dintr-un fișier text dat ca argument, efectuați

operațiunile solicitate, salvați datele din baza de date și generați rapoarte în format pdf.

***Analiza problemei***

Utilizatorul trevuie sa deschida Command Prompt (Consola), scriind in campul care contine calea cmd , apoi trebuie sa apese Enter.



Utilizatorul trebuie sa tasteze comada urmatoare: OrderManagement.jar < input file >

Comenzile suportate de aplicatie sunt:

* Insert client: prenume nume, adresa - In cazul in care clientul exista in baza de date nu se insereaza un alt client, pot exista clienti cu acelasi nume, dar strada diferita
* Insert product: nume, cantitate, pret - In cazul in care produsul exista in stoc, se actualizeaza stocul. Daca produsul existent in baza de date are un pret diferit, pretul acestuia este actualizat.
* Exemplu:
* Insert product: apple, 20, 1
* Insert product: apple, 10, 3
* In baza de date produsul este : apple, 30, 3
* Delete client: prenume nume, adresa - Se sterge atat clientul cat si comenzile facute de acesta.
* Delete Product: nume – Se sterge produsul din tabela produs si din toate comenzile
* Order: prenume nume, nume\_produs, cantitate – In baza de date este creata o comanda noua sau produsul este adaugat comenzii existente a clientului. De asemenea se genereaza o factura i format pdf, factura contine id\_ul facturi, numele si adresa clientului, data a care a fost creata, un tabel cu detalii referitoare la produsele comandate ( nume produs, cantitate, pret ). La finalul pdf-ului este afisat pretul total al comenzii.

Bill 51 Order date: Wed Apr 15 18:58:15 EEST 2020

Client Name: Luca George Address: Bucuresti

|  |  |  |
| --- | --- | --- |
| Product | Quantity | Price |
| apple | 5 | 1.0 |
| lemon | 5 | 2.0 |

Total Price: 15.0

In cazul in care cantitatea de produse pe care clientul doreste sa o comande este mai mica descat stocul depozitului este generat un pdf de eroare, comanda nu este procesata.

Error

Product apple stock is: 35

Your demand is 100

* Report client - Genereaza un pdf cu toti clientii din baza de date, echivalent cu SELECT \*

Client report

Wed Apr 15 18:58:14 EEST 2020

|  |  |  |
| --- | --- | --- |
| first\_name | last\_name | address |
| Luca | George | Bucuresti |
| Sandu | Vasile | Cluj-Napoca |

* Report product - Genereaza un pdf cu toate produsele din baza de date, echivalent cu SELECT \*

Product report

Wed Apr 15 18:58:15 EEST 2020

|  |  |  |
| --- | --- | --- |
| name | quantity | price |
| apple | 40 | 1.0 |
| orange | 40 | 1.5 |
| lemon | 70 | 2.0 |

* Report order - Genereaza un pdf cu toate comenzile din baza de date, pentru fiecare comanda se afiseaza urmatoarele informatii: Nume si adresa client, numarul facturii, data comenzii si pretul total al comenzii.

Order Report

Bill 52 Order date: Wed Apr 15 19:01:00 EEST 2020

Client Name: Luca George Address: Bucuresti

|  |  |  |
| --- | --- | --- |
| Product | Quantity | Price |
| apple | 115 | 1.0 |
| lemon | 115 | 2.0 |

Total Price: 345.0

Bill 53 Order date: Wed Apr 15 19:46:35 EEST 2020

Client Name: Sandu Vasile Address: Cluj-Napoca

|  |  |  |
| --- | --- | --- |
| Product | Quantity | Price |
| apple | 804 | 1.0 |
| lemon | 7 | 2.0 |
| orange | 9 | 1.5 |

Total Price: 831.5

***Proiectare***

*Am folosit desing pattern-ul* Layered Architecture, aplcatia contine pachetele

* ModelClasses – Contine clasele mapate in baza de date
* DataAccessClasses – Contine acele clase in care se face legatura cu baza de date
* PresentationClasses – Contine clasele care definesc interfata utilizatorului, fisiere de intrare si fisiere de iesire
* BussinesLogicClasses – Contine acele clase care incapsuleaza logica aplicatieai

