



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE**

TEMA

la disciplina

Tehnici de programare fundamentale

Order management

Borbei Raul

An academic: 2020 – 2021

Grupa: 30225



CUPRINS

1. OBIECTIVUL TEMEI	3
OBIECTIV PRINCIPAL	3
OBIECTIVE SECUNDARE.....	3
2. ANALIZA PROBLEMEI, MODELARE SCENARIU, CAZURI DE UTILIZARE	5
3. PROIECTARE (DECIZII DE PROIECTARE, DIAGrame UML, STRUCTURI DE DATE, PROIECTARE CLASE, INTERFEȚE, RELAȚII, PACKAGES, ALGORITMI, INTERFAȚA UTILIZATOR)	6
3.1. DECIZII DE PROIECTARE	6
3.2. DIAGRAMA UML.....	7
3.3. STRUCTURI DE DATE.....	8
3.4. INTERFAȚA UTILIZATOR	8
4. IMPLEMENTARE	12
5. REZULTATE	14
6. CONCLUZII.....	19
7. BIBLIOGRAFIE	20



1. Obiectivul temei

Obiectiv principal

Obiectivul principal al acestei teme de laborator este de a propune, proiecta și implementa o aplicație pentru gestionarea unei baze de date care conține clienți, produse și comenzi. Aplicația va conține trei interfețe grafice: una pentru clienți care va conține operațiile specifice, una pentru produse cu operațiile specifice și una pentru comenzi, care să permită crearea unei comenzi care să conțină un produs cu cantitatea dorită și un client care a comandat produsul, toate acestea fiind deja în baza de date aferentă aplicației. Aplicația trebuie să fie structurată pe pachete și să conțină minim următoarele: Model classes – pentru a stoca clasele din aplicatiei care se ocupa de model, Bussines Logic classes – care să conțină clasele ce se ocupa cu logica aplicației, Presentation classes – va conține clasele ce se ocupa de interfața grafica a aplicației și Data acces classes – va conține clasele pentru accesarea și lucrul cu baza de date.

Obiective secundare

Obiectivele secundare reprezintă pașii care trebuie urmați pentru a îndeplini obiectivul principal, iar ele sunt definite în cele ce urmează.

- Analiza problemei și identificarea cerințelor

Aplicația o dată pornită trebuie să îi permită utilizatorului să aleagă între cele 3 interfețe: client, produse sau comenzi. Dacă utilizatorul accesează interfața de client el trebuie să poată efectua operațiile specifice pe clienți și anume: să adauge un nou client în baza de date a aplicației, să editeze datele unui client deja existent în baza de date, să șteargă un client din baza de date și să poată vizualiza un tabel cu toți clienții din baza de date. Aceleași operații sunt posibile și pentru interfața produs. Dacă utilizatorul alege interfața comenzi el trebuie să poată crea o comandă nouă pentru un client deja existent în baza de date și să introducă un produs din baza de date. Interfața trebuie să fie una cât mai ușor de folosit de către utilizatori. Capitolul 2 prezintă în detaliu aceste obiective.

- Proiectarea aplicației



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE**

Ca date de intrare putem identifica datele introduse de către utilizator pentru fiecare tip de operație în parte. Aplicație trebuie să preia aceste date, să le transforme în tipurile necesare și să efectueze cu ele operația cerută asupra bazei de date existente. Detaliile obiectivului se găsesc în capitolul 3.

- Implementarea aplicației de gestiune

Cele trei ferestre din interfața grafică corespund tabelelor din baza de date și, de asemenea, ele constituie partea de model a aplicației noastre. Deci pentru început vom avea cele 3 clase necesare: Client, Product și Order, fiecare cu câmpurile aferente. Vom avea de asemenea nevoie de o clasă pentru conectarea la baza de date și una pentru efectuarea operațiilor asupra bazei de date. Vom avea nevoie de 4 clase pentru a crea interfața grafică și în final va fi nevoie de o clasă care să reprezinte logica aplicației și care va lega operațiile pe baza de date cu interfața grafică. Aceste clase sunt descrise în detaliu în capitolul 4.

- Testarea aplicației

Pentru testarea aplicației am populat baza de date cu 100 de clienți și cu 40 de produse cu cantitățile aferente. Am efectuat operațiile cerute pe câteva valori aleatoare și am observat comportamentul aplicației. Rezultatele vor apărea în capitolul 5 al documentației.



2. Analiza problemei, modelare scenarii, cazuri de utilizare

Cerința principală a temei este de a implementa o aplicație care să faciliteze un sistem de management pentru un magazin. Utilizatorul aplicației va putea astfel să își gestioneze lista de clienți și lista de produse și să creeze în orice moment o comandă pentru orice client din baza sa de date.

Scenariile de utilizare sunt diferite în funcție de ce dorește utilizatorul să facă. Dacă utilizatorul dorește să facă operații pe clienții acestora va porni aplicația și va apăsa pe butonul client din interfața principală. Interfața principală se va închide și va apărea în schimb interfața de client. În cazul în care utilizatorul dorește să adauge un nou client în baza de date va trebui să completeze câmpurile din interfață: id, name, address și phone nr. și să apese pe butonul „Add new client”, noul client va fi astfel adăugat în baza de date. Pentru a edita un client existent, utilizatorul va introduce id-ul clientului cărui dorește să îi modifice informațiile și în celelalte câmpuri va introduce noile informații pe care dorește să le aibă clientul, după care va apăsa pe butonul „Edit client”. Dacă se dorește ștergerea unui client din baza de date se va introduce id-ul clientului respectiv și se va apăsa pe butonul „Delete client”. Pentru a vizualiza clienții din baza de date se poate apăsa pe butonul „View clients” care va deschide o altă fereastră cu tabela de clienți. Operațiile pe produse sunt asemănătoare cu cele pe client, atât doar că avem la câmpuri id, name și quantity. În cazul în care utilizatorul dorește să creeze o nouă comandă va apăsa pe butonul „Order” care va deschide interfața aferentă și în plus va deschide o interfață cu tabelul de clienți și una cu tabelul de produse. În acest moment utilizatorul trebuie doar să aleagă ce produs dorește să plătească și să introducă id-urile aferente în câmpurile din interfață, să pună cantitatea dorită și să apese pe butonul „Add new order”. În cazul în care câmpurile nu sunt completate în momentul apăsării butonului va apărea un mesaj care îl roagă pe utilizator să introducă datele. În cazul în care câmpurile sunt completate corect dar cantitatea cerută depășește cantitatea disponibilă a produsului va apărea o eroare care îl va avertiza pe utilizator de asta. Altfel, dacă totul este bine noua comandă va fi plasată, cantitatea produsului de pe stoc va fi scăzută și se va genera o chitanță cu numele de Bill.txt în care se pot urmări ce s-a cumpărat, la ce moment de timp și la ce adresă trebuie să fie livrat.



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE**

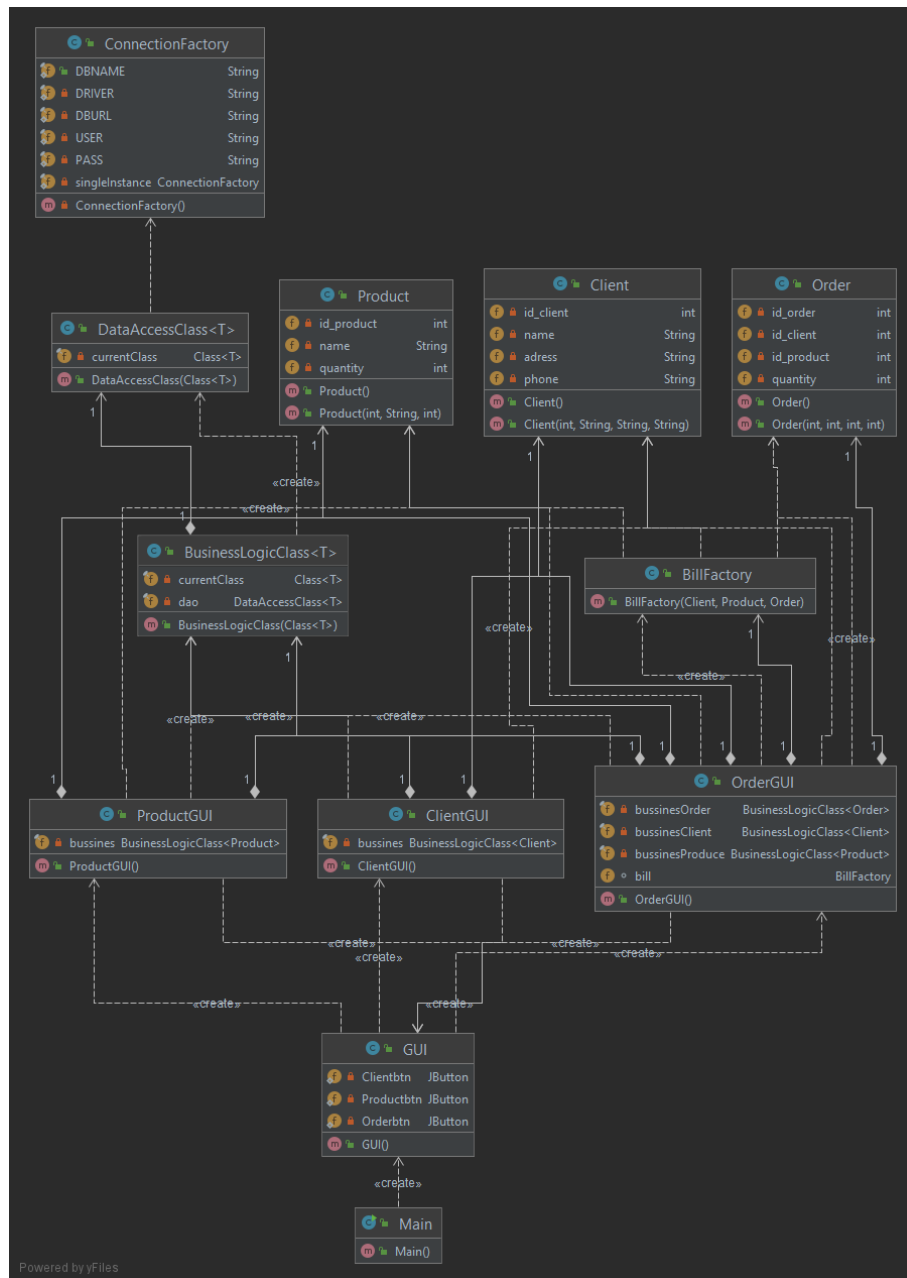
3. Proiectare (decizii de proiectare, diagrame UML, structuri de date, proiectare clase, interfețe, relații, packages, algoritmi, interfața utilizator)

3.1. Decizii de proiectare

Pentru implementarea aplicației vom avea nevoie în primul rând de clasele ce vor reprezenta modelul aplicației: Client, Product și Order. Aceste 3 clase vor avea aceleași nume de attribute ca și coloanele tabelor respective din baza de date. Aplicația mai are nevoie și de o clasă ConnectionFactory care va crea conexiunea cu baza de date. Pentru partea de interfață aplicația are încă 4 clase: una principala și celelalte 3 corespund fiecărei clase din tipul model. Pe urma aplicația mai are o clasă care se ocupa cu accesarea datelor din baza de date. În plus aplicația va mai conține o clasă pentru crearea unei chitanțe și o clasă de BussinesLogic care va lega interfața de baza de date prin intermediul clasei DataAcces.



3.2. Diagrama UML



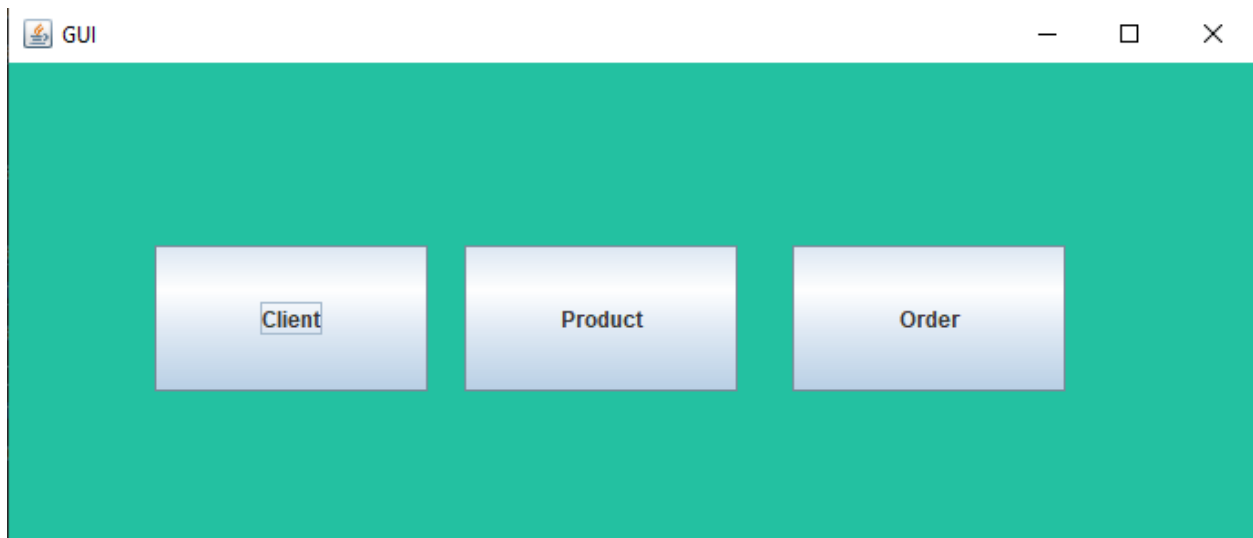


3.3. Structuri de date

În cadrul proiectului am folosit structuri de date de tipul `ArrayList<T>` în care am pus obiectele rezultate după interogarea bazei de date. `ArrayList` a fost transmis mai departe la interfața grafică sub forma unui `JTable` generat folosind tehnici de reflexie.

3.4. Interfața utilizator

La rularea aplicației utilizatorul va fi întâmpinat de următoarea interfață:



Este simplu de aici să își dea seama că în funcție de ce operații dorește să facă trebuie să selecteze butonul aferent.

Dacă se selectează butonul `Client` aplicația va deschide următoarea interfață grafică:

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE

Client

ID:

Name:

Adress:

Phone nr:

Add new client

Edit client

Delete client

View clients

BACK

În cadrul acestei interfețe utilizatorul poate sa introducă datele dorite sa și apese pe butonul corespunzător operației dorite. Pentru a efectua operațiile de add și edit toate câmpurile din interfața trebuie sa fie completate, iar pentru efectuarea operației de delete este de ajuns sa fie completat doar id-ul clientului care se dorește șters.

Daca utilizatorul va apasă butonul de „View clients” se va deschide alături o noua interfața care va conține un tabel cu toții clienții din baza de date.

Clients Table

id_client	name	adress	phone_nr
1	Sydnee Nielsen	Ap #212-9204 Risus. Av.	07 83 90 39 37
2	Keely Rowland	2665 Lacus St.	07 71 64 08 04
3	Octavia Finley	P.O. Box 644, 3455 Libe...	05 32 25 88 36
4	Anthony Bell	303-6663 Etiam Street	04 01 57 99 95
5	Hiram Whitley	7992 Quisque Road	05 37 15 14 40
6	Echo May	Ap #431-4058 Senectus...	06 80 34 10 18
7	Pascale Shepard	3317 Lacinia. Rd.	01 84 61 36 90
8	Moses Gutierrez	Ap #369-2251 Praesent ...	02 00 19 03 58
9	Gisela Strong	Ap #131-591 Magna. St.	08 05 50 38 65
10	Echo Perez	309-319 Dui. Street	04 25 30 62 91
11	Wylie Calhoun	538-3344 Et Road	02 14 93 93 29
12	Theodore Baldwin	Ap #929-4915 Sed Aven...	08 14 55 08 61
13	Armand Lowe	474-5008 Risus. Road	05 89 34 41 70
14	Macaulay Porter	343-6161 Pellentesque ...	07 34 41 68 84
15	Eaton Alvarez	Ap #429-7180 Urm Rd.	06 16 76 30 53
16	Uriel Roberts	Ap #467-8197 Commod...	03 90 87 55 62
17	Jillian Peck	111-3548 Lacinia. Ave	04 74 42 87 03
18	Madeline Workman	P.O. Box 890, 7359 Eni...	04 25 52 88 89
19	Leigh Benson	Ap #457-8513 Mattis. Rd.	01 74 72 46 48
20	Dakota Daniels	P.O. Box 483, 7113 Dolo...	08 45 28 01 82
21	Trevi Foley	340-2702 Pellentesque	03 02 02 13 54

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE

Product

ID:

Name:

Quantity:

Add new product

Edit product

Delete product

View products

BACK

Interfața Product este foarte asemănătoare cu cea de Client, singura diferență fiind ca în loc de address și phone_nr. utilizatorului i se cere cantitatea produselor din stoc.

De asemenea metodele sunt la fel și prin apăsarea butonului de „View products” va apărea un tabel cu produsele din stoc:

Products Table

id_product	name	quantity
1	Skittles	300
2	Red Vines	500
3	Reeses pieces	1200
4	Gummi Bears	300
5	Subway	120
6	Pizza Hut	2500
7	Pear	1000
8	Orange	200
9	Poppy seeds	1110
10	Havarti cheese	2500
11	Cinnamon Bread	1000
12	Cream Cheese Frosting	3200
13	Mango Lassi	320
14	Carbonara	1200
15	Pineapple	600
16	Fan	352
17	Alarm clock	125
18	Pocket knife	35
19	Sword	568
20	Basket	253
21	Skateboard	225

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE

Interfața Order este puțin diferită deoarece în momentul apăsării butonului Order în cadrul interfeței principale se va deschide o fereastră pentru plasarea unei comenzi și două ferestre ce vor conține tabelele cu clienții și cu produsele din baza de date. Utilizatorul va alege id-urile clientului și produsului dorit și va apăsa pe butonul Add pentru a plasa comanda. Comanda va fi plasată doar dacă în stoc sunt destule produse, în caz contrar va apărea un mesaj de atenționare.

The screenshot displays the 'Order' application interface. On the left, the 'Order' window contains input fields for 'ID client:', 'ID product:', and 'Quantity:', along with 'Add new order' and 'BACK' buttons. On the right, two data tables are shown:

Clients Table

id_client	name	adress	phone_nr
1	Sydnee Nielsen	Ap #212-9204 Risus. Av.	07 83 90 39 37
2	Keely Rowland	2665 Lacus St.	07 71 64 08 04
3	Octavia Finley	P.O. Box 644, 3455 Libe...	05 32 25 88 36
4	Anthony Bell	303-6663 Etiam Street	04 01 57 99 95
5	Hiram Whitley	7992 Quisque Road	05 37 15 14 40
6	Echo May	Ap #431-4058 Senectus...	06 80 34 10 18
7	Pascale Shepard	3317 Lacinia. Rd.	01 84 61 36 90
8	Moses Gutierrez	Ap #369-2251 Praesent...	02 00 19 03 58
9	Gisela Strong	Ap #131-591 Magna. St.	08 05 50 38 65
10	Echo Perez	309-319 Dui. Street	04 25 30 62 91
11	Wylie Calhoun	538-3344 Et Road	02 14 93 93 29
12	Theodore Baldwin	Ap #929-4915 Sed Aven...	08 14 55 08 61
13	Armand Lowe	474-5008 Risus. Road	05 89 34 41 70
14	Macaulay Porter	343-6161 Pellentesque...	07 34 41 68 84
15	Eaton Alvarez	Ap #429-7180 Urna Rd.	06 16 76 30 53
16	Uriel Roberts	Ap #467-8197 Commod...	03 90 87 55 62
17	Jillian Peck	111-3548 Lacinia. Ave	04 74 42 87 03
18	Madeline Workman	P.O. Box 890, 7359 Eni...	04 25 52 88 89
19	Leigh Benson	Ap #457-8513 Mattis. Rd.	01 74 72 46 48
20	Dakota Daniels	P.O. Box 483, 7113 Dolo...	08 45 28 01 82
21	Tonia Foley	240-2703 Pellentesque...	09 09 09 43 54

Products Table

id_product	name	quantity
1	Skittles	300
2	Red Vines	500
3	Reeses pieces	1200
4	Gummi Bears	300
5	Subway	120
6	Pizza Hut	2500
7	Pear	1000
8	Orange	200
9	Poppy seeds	1110
10	Havarti cheese	2500
11	Cinnamon Bread	1000
12	Cream Cheese Frosting	3200
13	Mango Lassi	320
14	Carbonara	1200
15	Pineapple	600
16	Fan	352
17	Alarm clock	125
18	Pocket knife	35
19	Sword	568
20	Basket	253
21	Crabapple	306



4. Implementare

După crearea bazei de date în sql și popularea sa cu câteva valori random pentru a putea fi testată aplicația ulterior am început să creez clasele din aplicației după cum urmează.

Am creat prima dată pachetul model care conține cele 3 clase: Client, Order și Product. Foarte important în crearea acestor clase a fost să mențin numele și ordinea câmpurilor precum tabelele corespunzătoare din baza de date.

Clasa Client va conține câmpurile `id_client` care este un identificator unic folosit în baza de date, `name` reprezentând numele clientului, `address` pentru adresa clientului și câmpul `phone_nr` în care reținem numărul de telefon al clientului.

Clasa Product va conține de asemenea un câmp de `id`, `id_product` în cazul acesta care va fi folosit pentru identificarea unică a produsului. De asemenea clasa va mai conține un field de `name`, pentru a putea introduce numele produsului și un field `quantity` în care vom reține cantitatea de pe stoc a produsului respectiv.

Clasa Order va conține un câmp de `id` propriu: `id_order` și pe urmă va conține un `id_client` și `id_product`. Aceste 2 câmpuri sunt folosite în baza de date ca și chei secundare pentru tabela Order. Pe lângă aceste câmpuri vom mai avea și un field `quantity` care ne va spune cantitatea comandată de clientul cu `id-ul=id_client` din produsul cu `id-ul=id_product`.

Toate aceste clase vor conține metode de `get` și `set` pentru a putea fi apelate în cadrul clasei care se ocupă cu comunicarea dintre baza de date și aplicația java.

Clasa `ConnectionFactory` este clasa care se ocupă cu crearea și închiderea conexiunii la baza de date. Aceasta conține câmpuri de tipul `private static final` în care se reține: `DBNAME`- numele bazei de date, `DRIVER`- driverul de `mysql`, `DBURL`-calea către baza de date, `USER`-numele userului din `mysql` și `PASS`-parola din `mysql`. Clasa va conține de asemenea și o instanță către ea înseși. Metodele din această clasa sunt implementate conform instrucțiunilor din prezentare și se folosesc pentru deschiderea și închiderea conexiunii la baza de date.

Clasa `DataAccessClass` este o clasă generică care conține o referință la clasa din care a fost apelată. În această clasă se află metodele pentru accesarea și manipularea datelor din baza de date. Avem în primul rând metoda `findAll()` implementată conform instrucțiunilor din prezentare care creează un statement de select folosind tehnici de reflexie, îl aplică pe baza de date și returnează rezultatul sub forma unui `ArrayList` de obiecte. Clasa este folosită în mod special pentru popularea tabelului de clienți și de produse. Metodele de `insert` și `update` creează instrucțiunile aferente și le

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE**

aplica pe baza de date folosind tehnici de reflexie pe un obiect generic. Metoda de delete creează interogarea de delete pe un id dat ca și parametru în cadrul tabelului aferent clasei din care a fost apelata. Metodele de getObjectData sunt folosite pentru a crea și apela metodele de tipul get ale obiectului pe care sunt apelate. În mod similar metoda setObjectData creează și apelează metoda de set. Metoda getId returnează obiectul cu id-ul dat ca și parametru din tabelul aferent din baza de date. Metoda getMaxId este folosită pentru generarea unei chitanțe pentru ca avem nevoie de id-ul următor când cream o nouă comandă.

Clasa BillFactory este responsabilă de crearea unei chitanțe. Constructorul acestei clase primește ca și parametru un obiect de tip client, unul de tip produs și unul de tip comandă și tipărește datele importante despre cele trei împreună cu data și timpul la care a fost generată chitanța.

Clasele din pachetul presentation reprezintă clasele responsabile de interfața grafică. Avem clasa GUI care este prima deschisă la rularea aplicației și din ea se pot accesa celelalte 3: ClientGUI, ProductGUI și OrderGUI. Ultimele 3 au implementat un buton de back care deschide interfața principală.

Clasa BusinessLogicClass este legătura dintre DataAcces și interfața grafică. Este o clasă generică ce conține o referință de DataAcces și una de clasă curentă. Această clasă conține metodele de insertRow, updateRow și deleteRow care se apelează prin apăsarea butoanelor din interfața grafică. De asemenea conține metoda updateTable care generează headerul unui tabel prin tehnici de reflexie și pe urmă îl populează cu datele corespunzătoare din baza de date.



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE CATEDRA CALCULATOARE

5. Rezultate

Pentru testarea aplicației am încercat toate operațiile din interfața grafică. Adaug aici un exemplu în care încerc să mă înserez pe mine ca și client pe lângă cei 100 deja existenți.

The screenshot displays two windows from a graphical user interface. The 'Client' window on the left has a teal background and contains input fields for ID, Name, Address, and Phone number, along with buttons for 'Add new client', 'Edit client', 'Delete client', 'View clients', and 'BACK'. The 'Clients Table' window on the right has a white background and displays a table of client data.

id_client	name	adress	phone_nr
81	Jemima Wilkins	4789 Placerat, Road	08 61 97 26 74
82	Ezekiel Oneill	746-1371 Dignissim Av...	08 75 18 05 44
83	Oliver Roberson	Ap #222-910 Praesent ...	07 50 27 99 02
84	Chase Ayers	6146 Sed Ave	05 62 89 84 37
85	Devin Pickett	686-5654 Malesuada St...	01 18 62 41 27
86	Demetria Greene	Ap #138-8156 Dui. Rd.	07 21 02 19 84
87	Kiara Henson	P.O. Box 781, 9408 Fusc...	04 84 29 34 67
88	Jolie Schultz	P.O. Box 793, 1830 Sed ...	05 74 99 45 55
89	Noelle Horne	6605 Metus. Rd.	06 23 31 65 52
90	Abel Knapp	6208 Elit. Av.	07 46 12 96 91
91	Raya Drake	P.O. Box 298, 2774 Tinci...	01 93 71 24 83
92	Joshua Delacruz	P.O. Box 591, 4326 Rutr...	04 80 34 26 81
93	Upton Hale	P.O. Box 689, 2399 Ege...	05 89 44 87 01
94	Neve Dudley	Ap #939-4619 Turpis. St.	06 20 69 66 58
95	Gray Cohen	Ap #327-2765 Arcu Aven...	01 32 25 20 70
96	Mohammad Hayes	5754 Auctor Rd.	06 32 37 97 41
97	Tucker Mcdaniel	Ap #316-4304 Aliquam ...	04 17 24 23 59
98	Amena Beck	Ap #314-5921 Parturient...	01 10 15 43 22
99	Noble Spears	307-7451 Nunc Av.	03 81 04 38 27
100	Dieter Monroe	P.O. Box 850, 4633 Nas...	09 78 57 70 70
101	Borbei Raul	Cluj Napoca, nr 3, ap 6	07 58 27 45 69

Stergem după aceea clientul cu id-ul 5 și mutăm clientul cu id-ul 6 în România:

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE

Client

ID:

6

Name:

Echo May

Adress:

Romania

Phone nr:

06 80 34 10 18

Add new client

Edit client

Delete client

View clients

BACK

Clients Table

id_client	name	adress	phone_nr
1	Sydney Nielsen	Ap #212-9204 Risus. Av.	07 83 90 39 37
2	Keely Rowland	2665 Lacus St.	07 71 64 08 04
3	Octavia Finley	P.O. Box 644, 3455 Libe...	05 32 25 88 36
4	Anthony Bell	303-6663 Etiam Street	04 01 57 99 95
6	Echo May	Romania	06 80 34 10 18
7	Pascale Shepard	3317 Lacinia. Rd.	01 84 61 36 90
8	Moses Gutierrez	Ap #369-2251 Praesent...	02 00 19 03 58
9	Gisela Strong	Ap #131-591 Magna. St.	08 05 50 38 65
10	Echo Perez	309-319 Dui. Street	04 25 30 62 91
11	Wylie Calhoun	538-3344 Et Road	02 14 93 93 29
12	Theodore Baldwin	Ap #929-4915 Sed Aven...	08 14 55 08 61
13	Armand Lowe	474-5008 Risus. Road	05 89 34 41 70
14	Macaulay Porter	343-6161 Pellentesque...	07 34 41 68 84
15	Eaton Alvarez	Ap #429-7180 Uma Rd.	06 16 76 30 53
16	Uriel Roberts	Ap #467-8197 Commod...	03 90 87 55 62
17	Jillian Peck	111-3548 Lacinia. Ave	04 74 42 87 03
18	Madeline Workman	P.O. Box 890, 7359 Eni...	04 25 52 88 89
19	Leigh Benson	Ap #457-8513 Mattis. Rd.	01 74 72 46 48
20	Dakota Daniels	P.O. Box 483, 7113 Dolo...	08 45 28 01 82
21	Travis Foley	249-2703 Pellentesque...	02 02 92 13 51
22	Karen Day	Ap #506-7760 Ultrices...	02 52 62 10 35

Putem observa ca aceste metode functioneaza. Aceleasi metode se aplica și în cazul clasei Product deci nu mai includ aici tot, în schimb voi adauga în stoc un produs nou.

Product

ID:

45

Name:

iMacPro

Quantity:

5

Add new product

Edit product

Delete product

View products

BACK

Products Table

id_product	name	quantity
26	Bed	1354
27	Table	1203
28	Floor lamp	130
29	Cutting board	256
30	Mug	456
31	Fork	852
32	Calculator	547
33	Canoe	587
34	Rocking chair	423
35	Vase	52
36	Flashlight	48
37	Spoon	8874
38	Glasses	458
39	Can opener	258
40	Bowl	544
41	Playing cards	88
42	Ring	77
43	Toy train	56
44	Frying pan	426
45	iMacPro	5

Se observa ca functioneaza și acum voi merge și voi comanda un produs în interfata Order.



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE

The screenshot displays a database application interface with the following components:

- Order Form (Left):** A form for creating a new order. It includes input fields for "ID client:" (value: 101), "ID product:" (value: 45), and "Quantity:" (value: 6). Below these fields are two buttons: "Add new order" and "BACK".
- Clients Table (Top Right):** A table listing client information. The columns are "id_client", "name", "address", and "phone_nr". The table contains 20 rows of data, with the last row (id 101) corresponding to the client selected in the order form.
- Products Table (Bottom Right):** A table listing product information. The columns are "id_product", "name", and "quantity". The table contains 20 rows of data, with the last row (id 45) corresponding to the product selected in the order form.
- Error Message (Center):** A modal dialog box titled "Message" with an information icon. It displays the text "Not enough products" and an "OK" button.

Observam ca nu pot comanda mai multe produse decat exista pe stoc. Schimb datele și comand un singur produs.

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE

id_client	name	adress	phone_nr
82	Ezekiel Oneill	746-1371 Dignissim Av...	08 75 18 05 44
83	Oliver Roberson	Ap #222-910 Praesent ...	07 50 27 99 02
84	Chase Ayers	6146 Sed Ave	05 62 89 84 37
85	Devin Pickett	686-5654 Malesuada St...	01 18 62 41 27
86	Demetria Greene	Ap #138-8156 Dui. Rd.	07 21 02 19 84
87	Kiara Henson	P.O. Box 781, 9408 Fusc...	04 84 29 34 67
88	Jolie Schultz	P.O. Box 793, 1830 Sed ...	05 74 99 45 55
89	Noelle Horne	6605 Metus. Rd.	06 23 31 65 52
90	Abel Knapp	6208 Elit. Av.	07 46 12 96 91
91	Raya Drake	P.O. Box 298, 2774 Tinci...	01 93 71 24 83
92	Joshua Delacruz	P.O. Box 591, 4326 Rutr...	04 80 34 26 81
93	Upton Hale	P.O. Box 689, 2399 Ege...	05 89 44 87 01
94	Neve Dudley	Ap #939-4619 Turpis. St.	06 20 69 66 58
95	Gray Cohen	Ap #327-2765 Arcu Aven...	01 32 25 20 70
96	Mohammad Hayes	5754 Auctor Rd.	06 32 37 97 41
97	Tucker Mcdaniel	Ap #316-4304 Aliquam ...	04 17 24 23 59
98	Amena Beck	Ap #314-5921 Parturient...	01 10 15 43 22
99	Noble Spears	307-7451 Nunc Av.	03 81 04 38 27
100	Dieter Monroe	P.O. Box 850, 4633 Nas...	09 78 57 70 70
101	Borbei Raul	Cluj Napoca, nr 3, ap 6	07 58 27 45 69

id_product	name	quantity
26	Bed	1354
27	Table	1203
28	Floor lamp	130
29	Cutting board	256
30	Mug	456
31	Fork	852
32	Calculator	547
33	Canoe	587
34	Rocking chair	423
35	Vase	52
36	Flashlight	48
37	Spoon	8874
38	Glasses	458
39	Can opener	258
40	Bowl	544
41	Playing cards	88
42	Ring	77
43	Toy train	56
44	Frying pan	426
45	iMacPro	4

Operatia se efectueaza cu succes și se poate observa ca stocul a scazut. De asemenea putem gasi chitanta în fisierul Bill.txt:



UNIVERSITATEA TEHNICĂ

DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
CATEDRA CALCULATOARE

```
1 19:00:53 2021/06/01
2
3 Order Number: 3
4 Product: iMacPro
5 Quantity: 1
6 Name: Borbei Raul
7 Address: Cluj Napoca, nr 3, ap 6
```

Rezultatele sunt satisfacatoare și aplicatia se comporta corespunzator.



6. Concluzii

În urma efectuării temei3 mi-am reamintit de conexiunile dintre sql și java, cum se creează ele și cum se obțin datele dintr-o baza de date astfel încât să poată fi folosite în cadrul unei aplicații Java. Programul a fost realizat folosind tehnici de programare orientate obiect. M-am familiarizat cu folosirea de clase generice și cu folosirea tehnicilor de reflectare.

Proiectul poate fi dezvoltat pe viitor prin înfrumusețarea interfeței grafice, care în momentul de față este foarte simplă. Ulterior pot fi implementate mai multe strategii de căutare între clienții. În mod special partea de comandă poate fi dezvoltată să permită de exemplu căutarea unui client folosind numele sau și la fel poate fi făcut și pentru produse.



7. Bibliografie

<https://www.generatedata.com/>

<https://www.jetbrains.com/help/idea/working-with-code-documentation.html#toggle-rendered-view>

<https://www.baeldung.com/javadoc>

https://www.codota.com/?utm_source=search-web

<https://www.jetbrains.com/help/idea/work-with-maven-dependencies.html>

<http://tutorials.jenkov.com/java-reflection/index.html>

https://gitlab.com/utc_n_dsrl/pt-layered-architecture

https://gitlab.com/utc_n_dsrl/pt-reflection-example