

Criteria B: Design

➤ Database Tables

- Table Name: branchlist

Field Name	Field Type	Properties	Description	Other Characteristics
BranchID	VARCHAR (80)	Not Null	<ul style="list-style-type: none">• Uniquely identifies each client.• Generated in Java and inserted in the database.• (BRANCH+(number of seconds since 1970))	<ul style="list-style-type: none">• Primary Key
Address	VARCHAR (250)	Not Null	<ul style="list-style-type: none">• Address of the branch	
ContactNumber	VARCHAR (15)	Not Null	<ul style="list-style-type: none">• Contact Number of the branch	

- Table Name: clientlist

Field Name	Field Type	Properties	Description	Other Characteristics
ClientID	VARCHAR (20)	Not Null	<ul style="list-style-type: none">• Uniquely identifies each client.• Generated in Java and inserted in the database.• (CLIENT+ (number of seconds since 1970))	<ul style="list-style-type: none">• Primary Key
Name	VARCHAR (20)	Not Null	<ul style="list-style-type: none">• Name of the client	
MobileNumber	VARCHAR (45)	Not Null	<ul style="list-style-type: none">• Contact Number of the client	<ul style="list-style-type: none">• Primary Key
Email	VARCHAR (45)		<ul style="list-style-type: none">• E-mail of the client	
Gender	CHAR (1)	Not Null	<ul style="list-style-type: none">• Gender of the client (M or F)	
Birthday	VARCHAR (10)	Not Null	<ul style="list-style-type: none">• Birthdate of client	

- Table Name: invoicelist

Field Name	Field Type	Properties	Description	Other Characteristics
SerialNo	INT (11)	Not Null	<ul style="list-style-type: none"> • Previous value increases by 1 whenever a new invoice is generated. 	<ul style="list-style-type: none"> • Primary Key • Auto Increment
Path	VARCHAR (300)	Not Null	<ul style="list-style-type: none"> • File path of the invoice generated is stored in the database (so that it can be used later to open the invoice from the list) 	
ClientName	VARCHAR (100)	Not Null	<ul style="list-style-type: none"> • Name of the client whose invoice is made is saved 	
Date	DATE	Not Null	<ul style="list-style-type: none"> • Date and Time of when the invoice is generated 	<ul style="list-style-type: none"> • Primary Key
Expense	DECIMAL (11,0)	Not Null	<ul style="list-style-type: none"> • Expense of the invoice 	

- Table Name: invoiceproductlist

Field Name	Field Type	Properties	Description	Other Characteristics
line_id	INT (11)	Not Null	<ul style="list-style-type: none"> • Row counter 	<ul style="list-style-type: none"> • Primary Key
invoiceno	INT (20)	Not Null	<ul style="list-style-type: none"> • Invoice serial number 	
Date	DATE	Not Null	<ul style="list-style-type: none"> • Date when the invoice was generated 	
Product Name	VARCHAR (250)	Not Null	<ul style="list-style-type: none"> • Name of the product 	
productid	INT (20)	Not Null	<ul style="list-style-type: none"> • Product ID 	
price	DECIMAL (20,0)	Not Null	<ul style="list-style-type: none"> • Price of the product 	

- Table Name: invoiceservicelist

Field Name	Field Type	Properties	Description	Other Characteristics
line_id	INT (11)	Not Null	<ul style="list-style-type: none"> • Row counter 	<ul style="list-style-type: none"> • Primary Key
invoiceno	INT (20)	Not Null	<ul style="list-style-type: none"> • Invoice serial number 	
Date	DATE	Not Null	<ul style="list-style-type: none"> • Date when the invoice was generated 	
Service Name	VARCHAR (250)	Not Null	<ul style="list-style-type: none"> • Service name 	
serviceID	INT (20)	Not Null	<ul style="list-style-type: none"> • Service ID 	
price	DECIMAL (20,0)	Not Null	<ul style="list-style-type: none"> • Service price 	

- Table Name: productlist

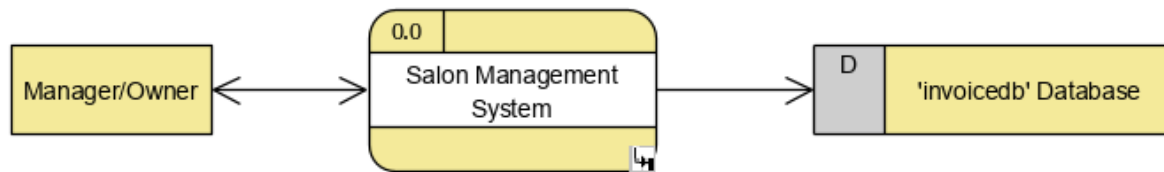
Field Name	Field Type	Properties	Description	Other Characteristics
SerialNo	INT (11)	Not Null	<ul style="list-style-type: none"> • Previous value increases by 1 whenever a new product is added to the list. 	<ul style="list-style-type: none"> • Primary Key • Auto Increment
Company	VARCHAR (45)	Not Null	<ul style="list-style-type: none"> • Company of the product 	
ProductName	VARCHAR (200)	Not Null	<ul style="list-style-type: none"> • Name of the product 	
ProductDescription	VARCHAR (200)		<ul style="list-style-type: none"> • Description of the product (can be left blank) 	
Size	DECIMAL (10, 0)	Not Null	<ul style="list-style-type: none"> • Size of the product (in mL) 	
Rate	DECIMAL (10, 0)	Not Null	<ul style="list-style-type: none"> • Unit price of the product (Price of 1 quantity) 	

- Table Name: servicelist

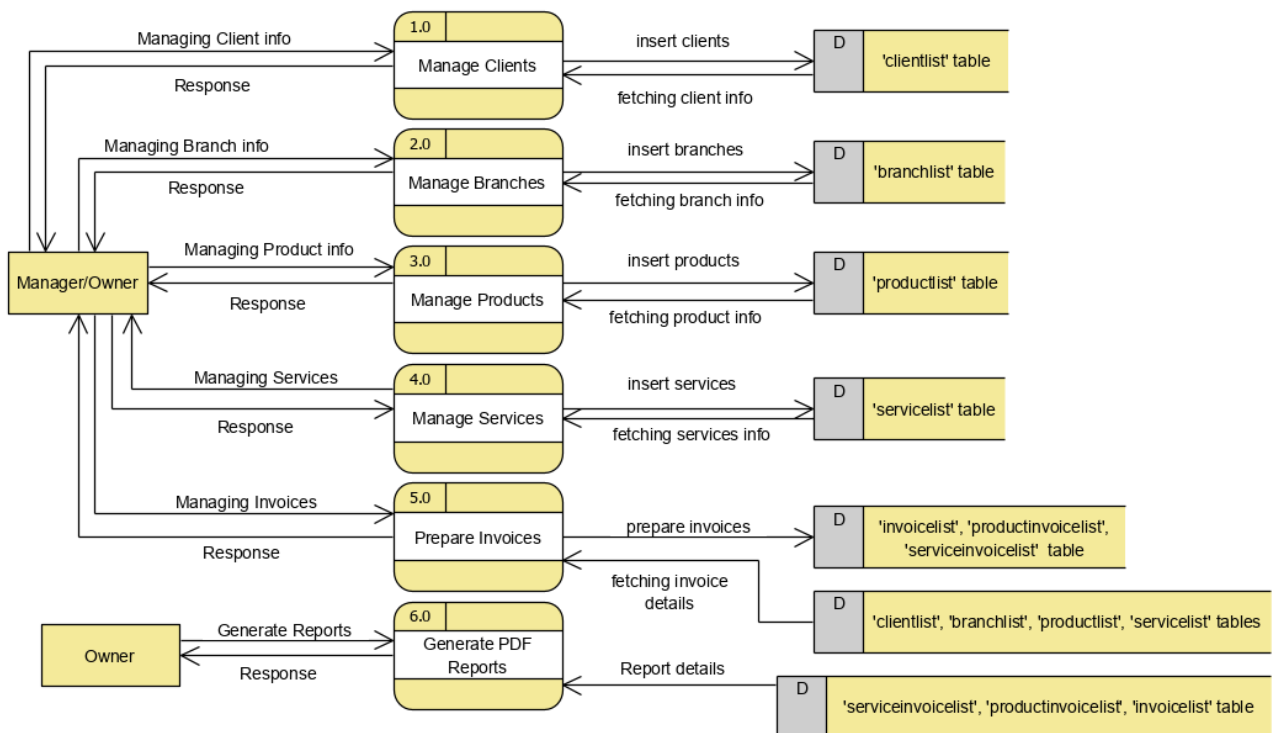
Field Name	Field Type	Properties	Description	Other Characteristics
SerialNo	INT (11)	Not Null	<ul style="list-style-type: none"> • Previous value increases by 1 whenever a new product is added to the list. 	<ul style="list-style-type: none"> • Primary Key • Auto Increment
Category	VARCHAR (45)	Not Null	<ul style="list-style-type: none"> • Category of the service (Example: hair, skin, etc.) 	
Subcategory	VARCHAR (45)	Not Null	<ul style="list-style-type: none"> • Sub-category of the service (For e.g.: in the category Hair, sub-category can be Haircut for Female/Haircut for Male) 	
Rate	DECIMAL (11, 0)	Not Null	<ul style="list-style-type: none"> • Unit price of the service (Price of 1 qty) 	

➤ Data Flow Diagram (DFD)

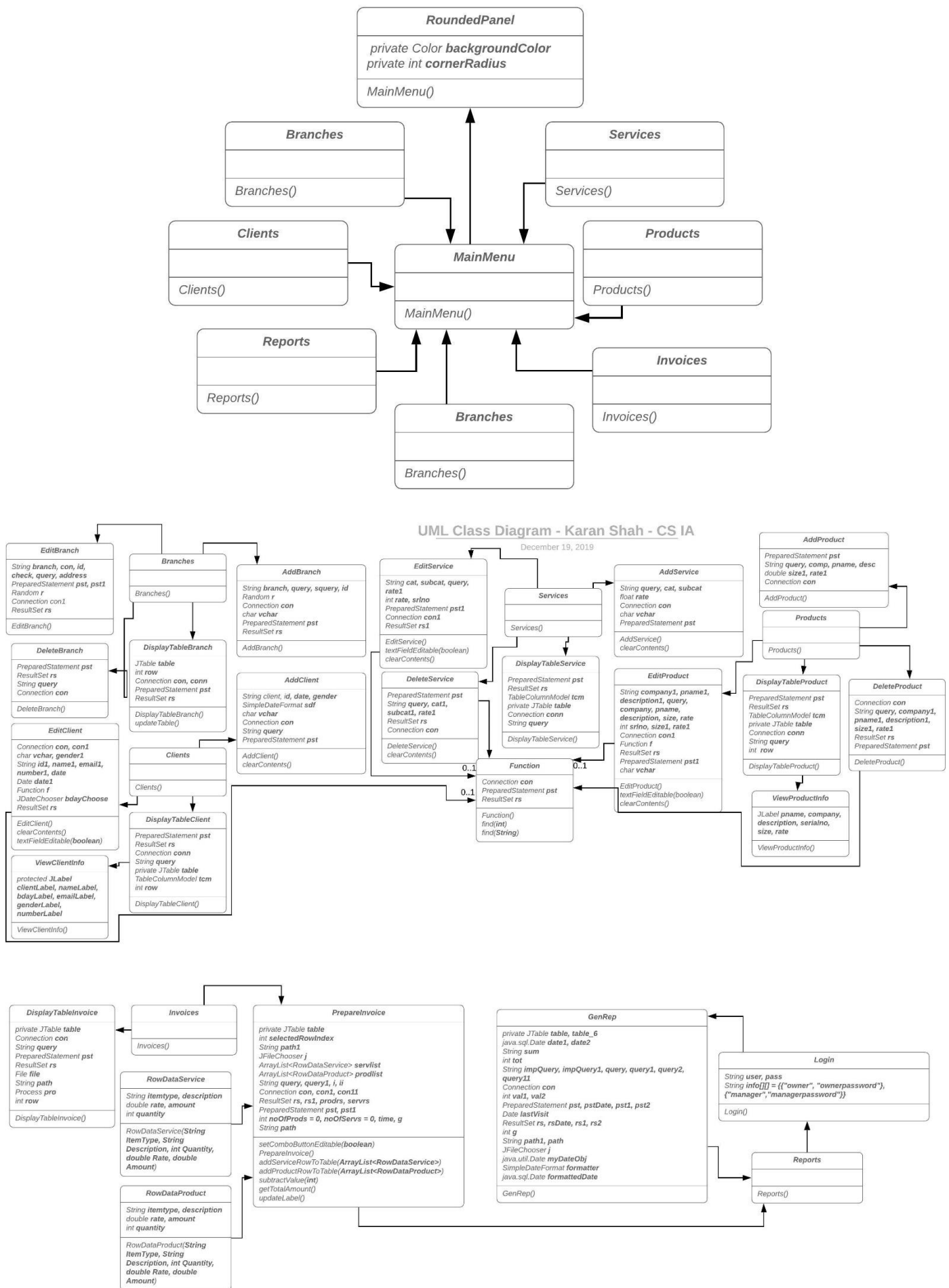
a. Data Flow Diagram Level 0



b. Data Flow Diagram Level 1



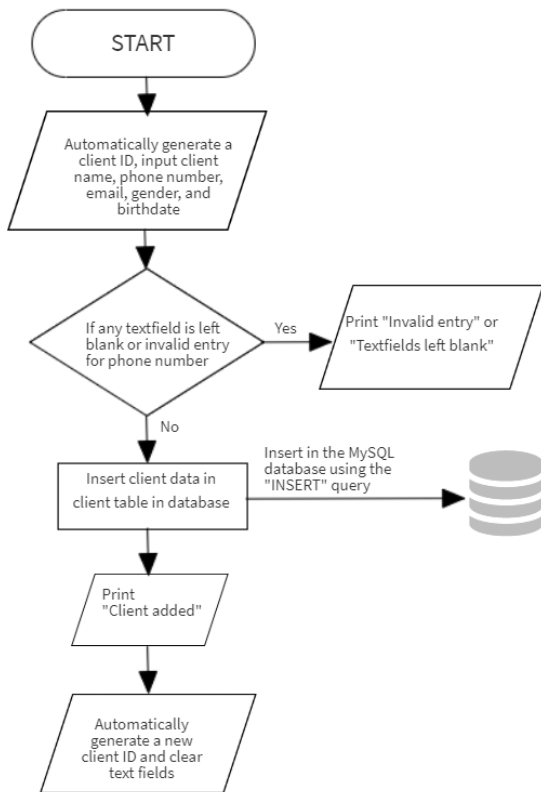
➤ UML Diagram



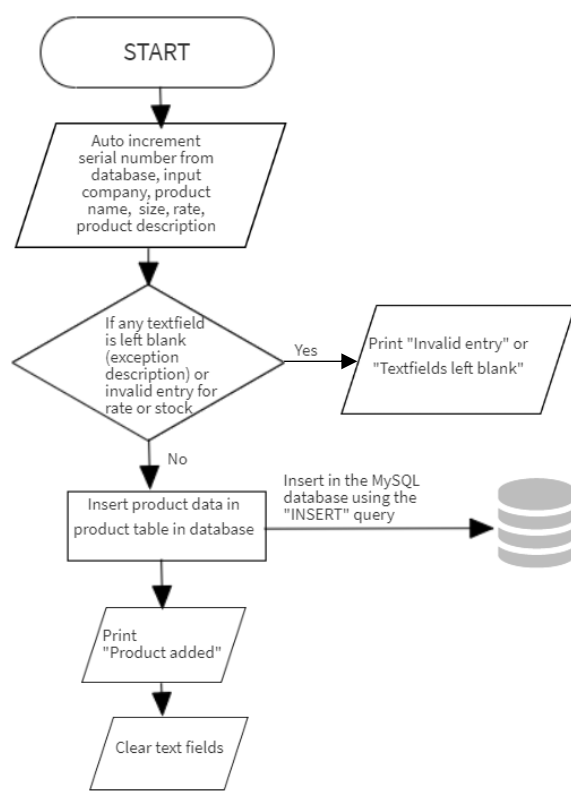
➤ **Flowcharts for all the processes in the program**

• **Inserting data into database**

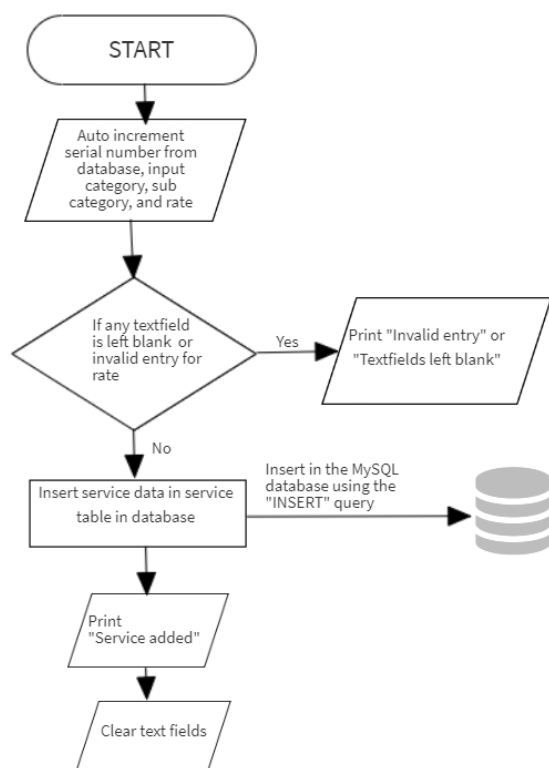
Adding a new client



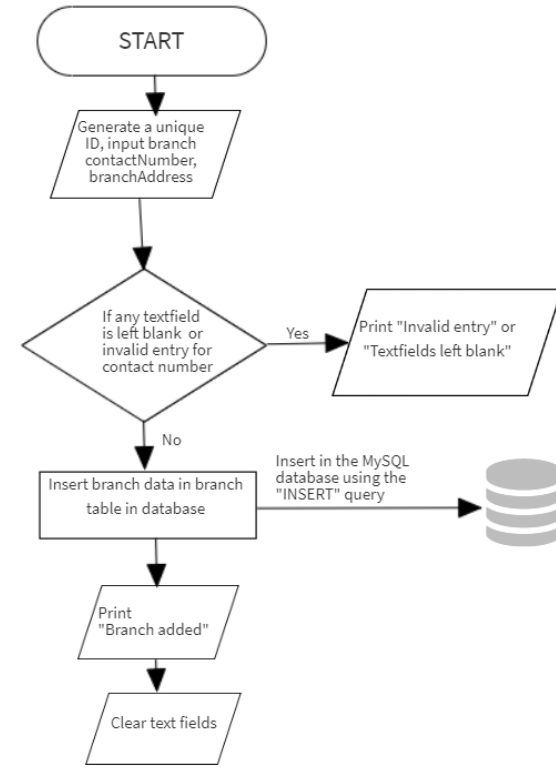
Adding a new Product



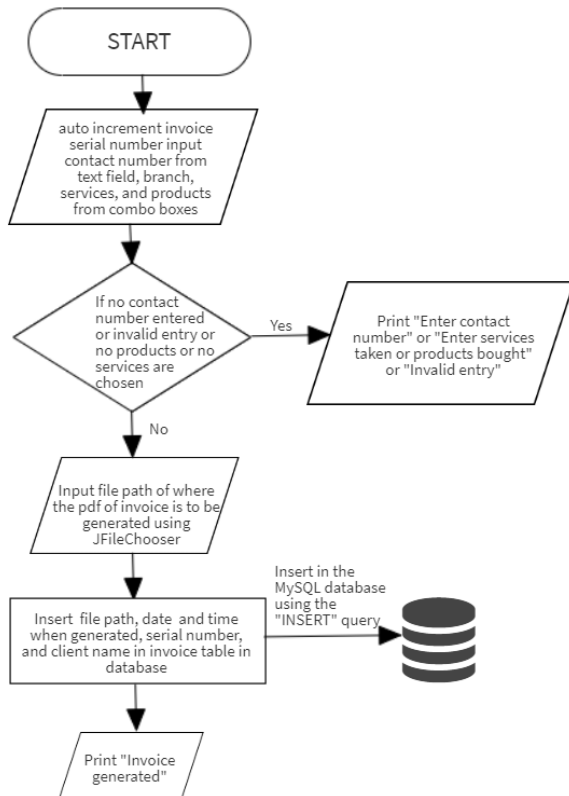
Adding a new Service



Adding a new Branch

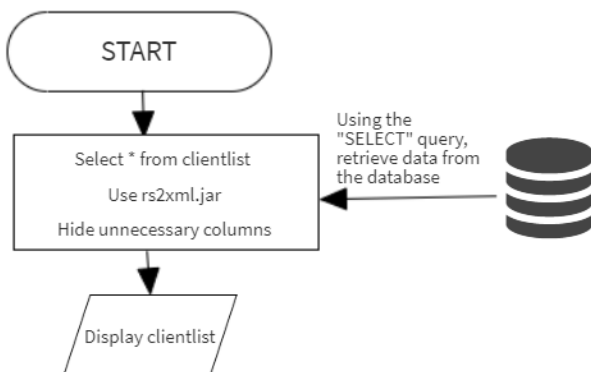


Making a new invoice

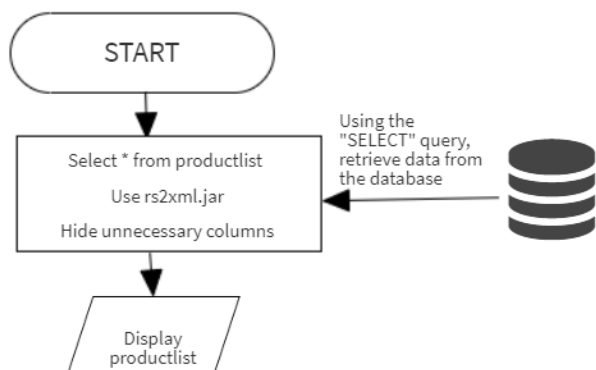


- **Displaying data to tables from databases**

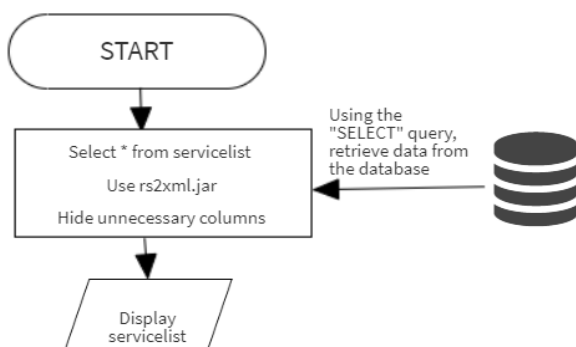
View client list from client table



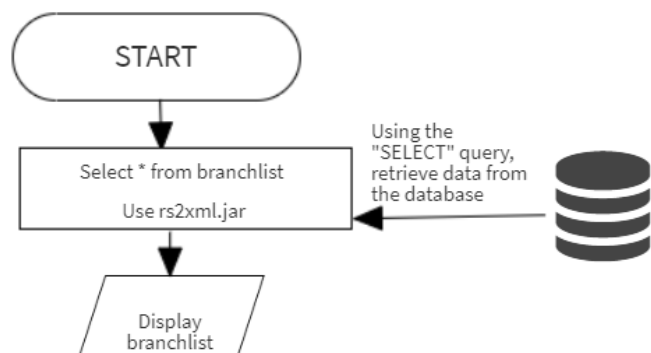
View product list from product table



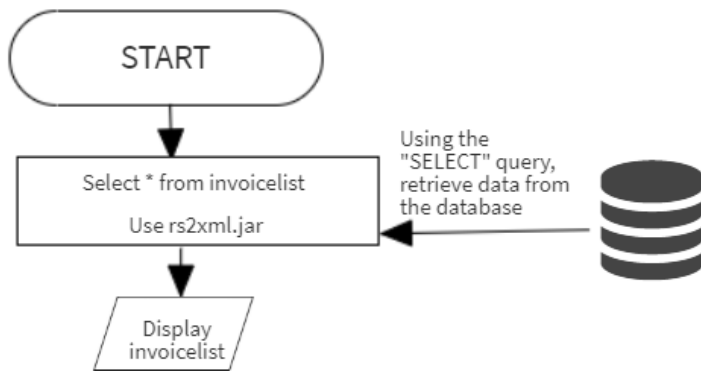
View services' list from service table



View branches' list from branches table

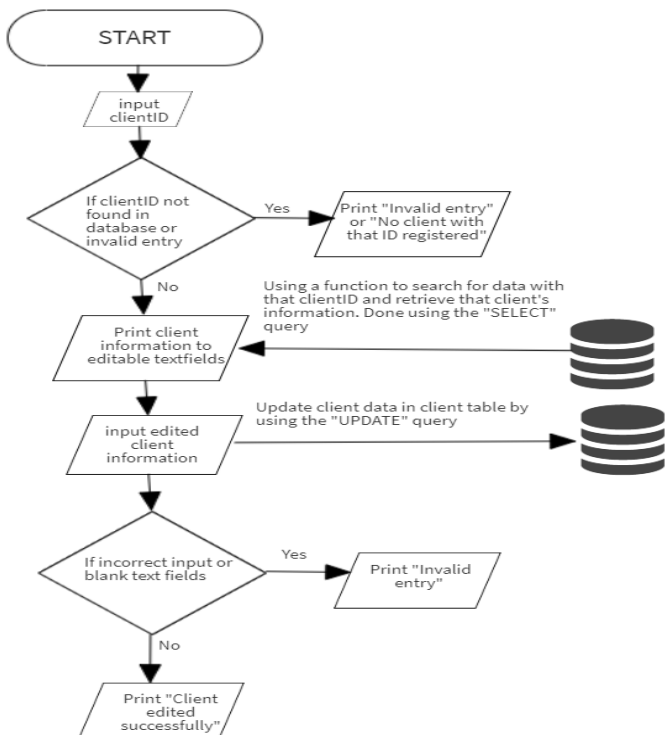


View invoice list from invoice table

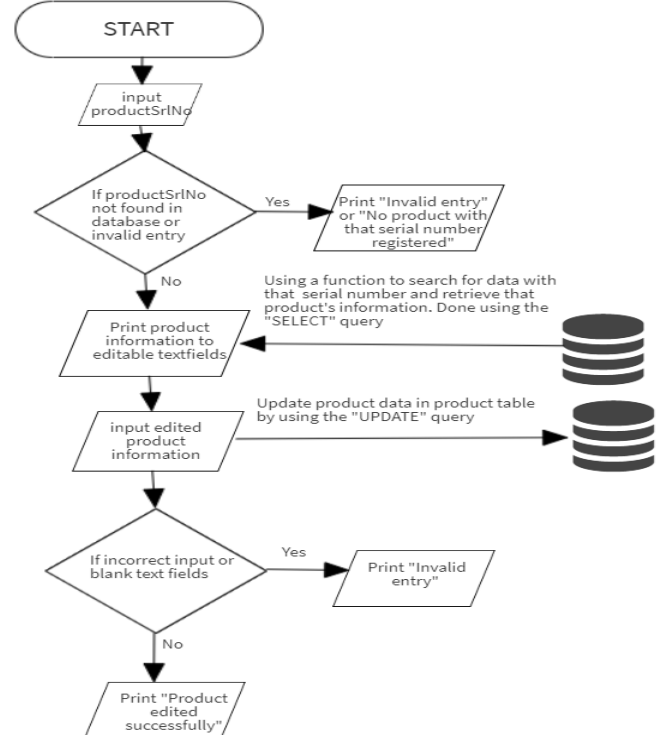


- Editing existing data and updating database

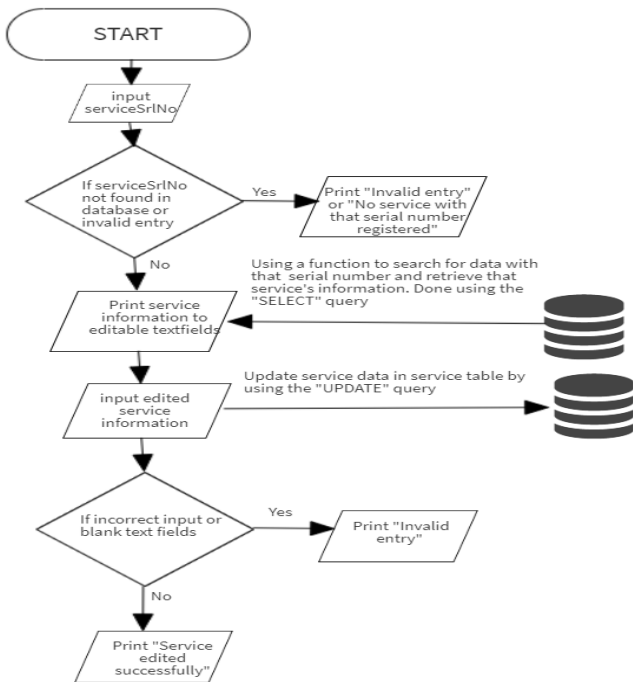
Edit client data



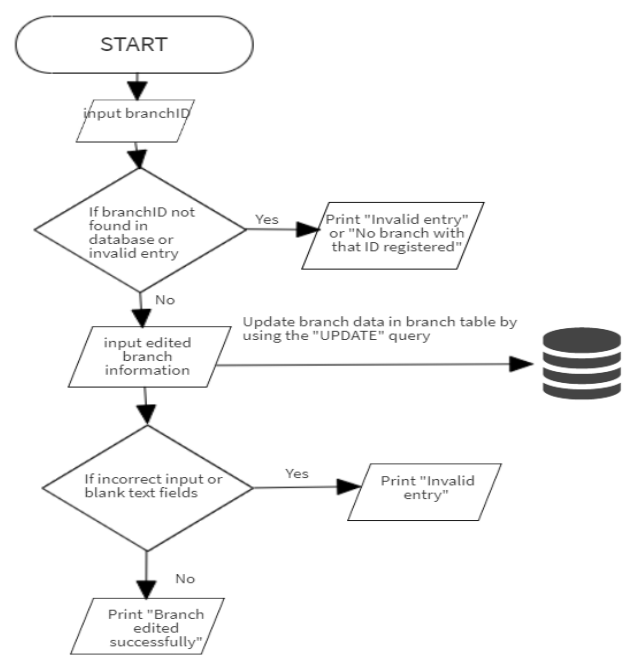
Edit product data



Edit service data

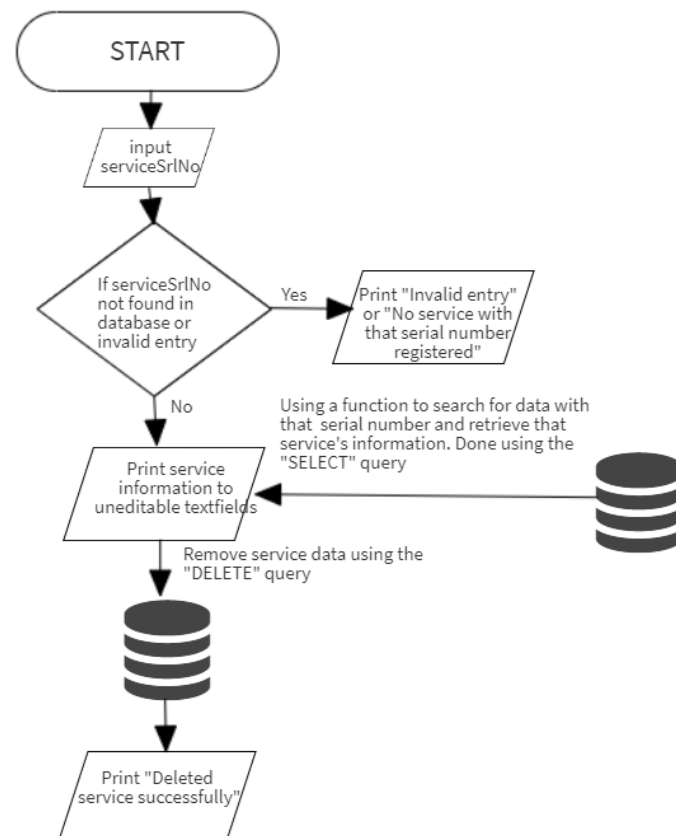


Edit branch data

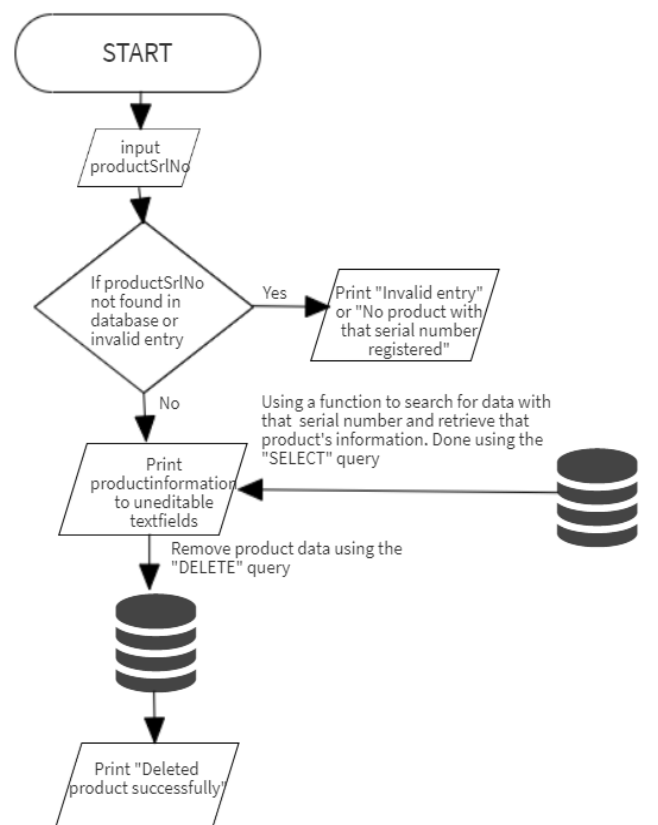


Deleting data from database

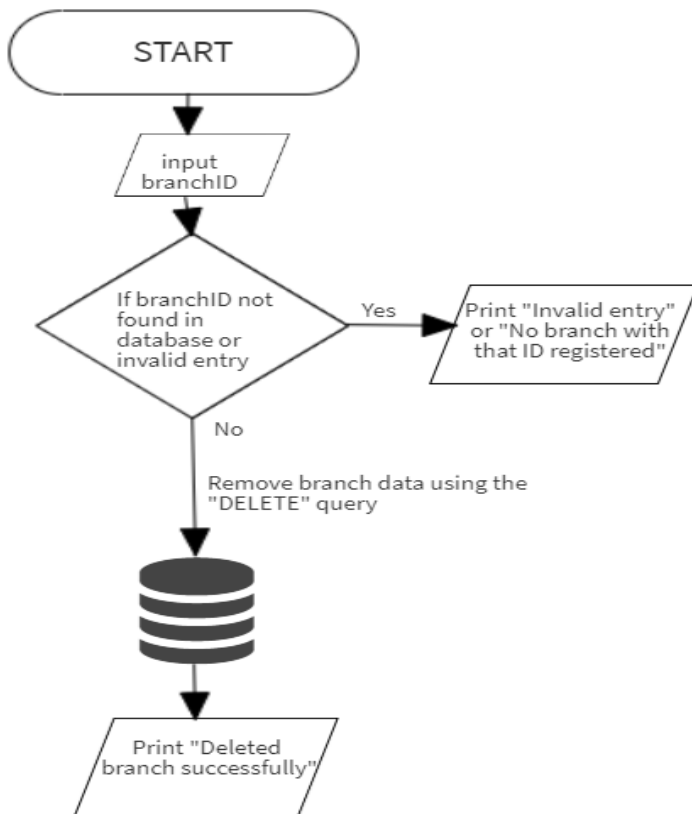
Removing a service



Deleting a product

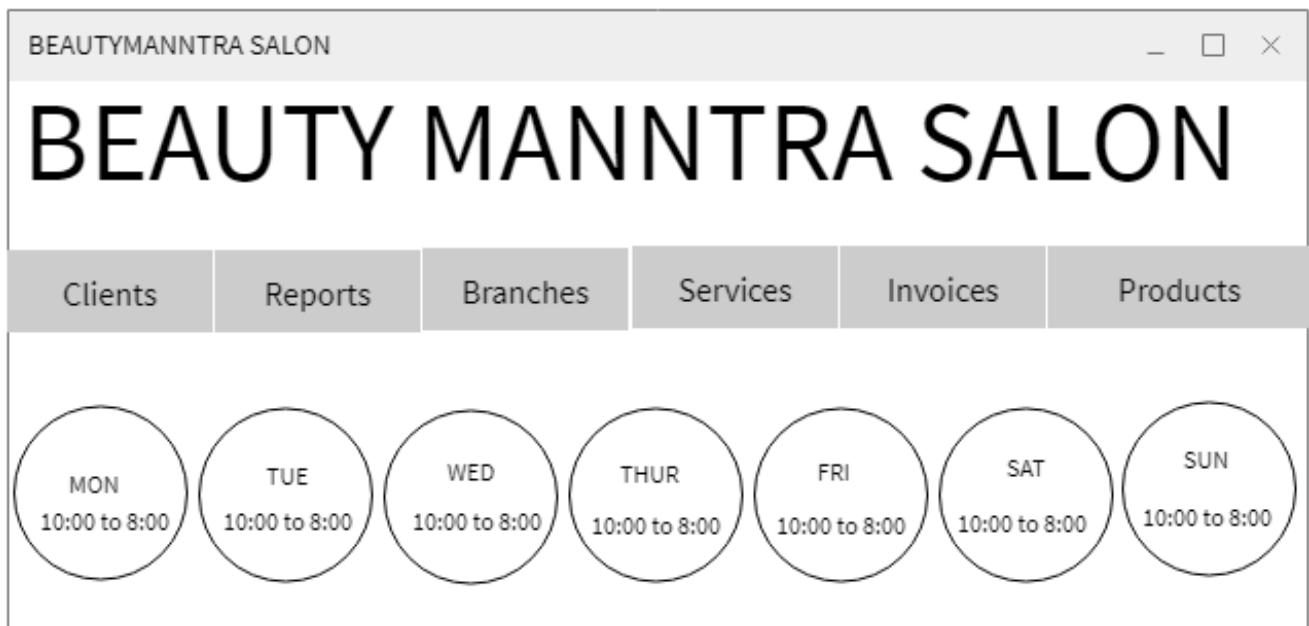


Deleting a branch



➤ GUI Design of the program

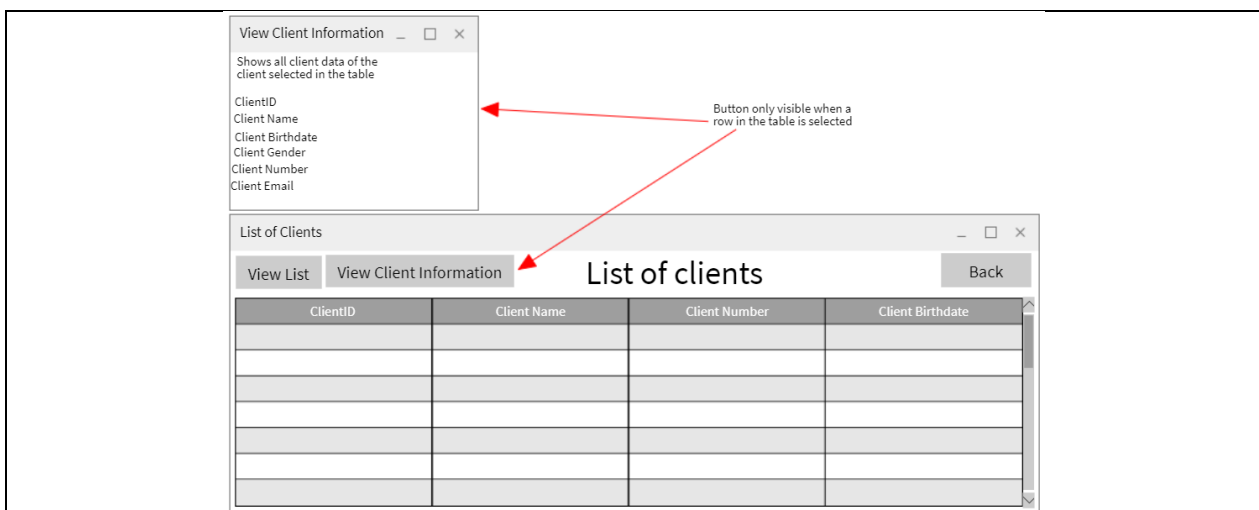
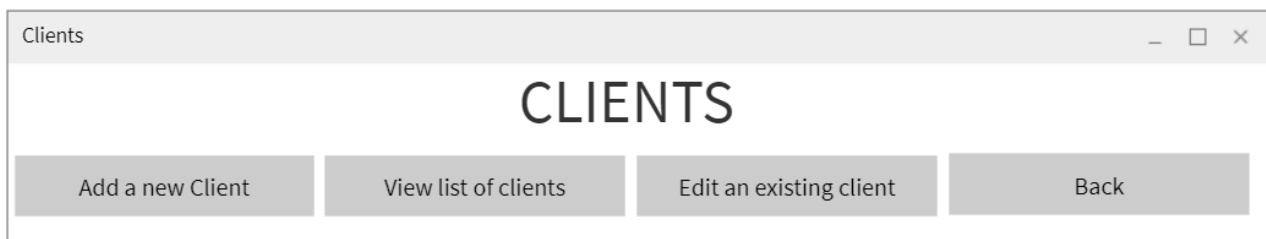
Main Menu of the program:



- 6 toggle buttons will be added to a button group (Clients, Products, Services, Invoice, Branch, Reports) so that only one of the buttons can be pressed at a time.
- Uses the class RoundedPanel.java to make the circles in which the days and times of when the beauty salon is open will be written. This matches one of the features of the website of the salon:

Sub-menus of the main menu:

1. Client Menu



2. Product Menu

Products

PRODUCTS

Add a new product

View products

Edit product

Delete product

Back

Add a client

ID: {automatically generated, e.g.: CLIENT1577647007}

Name:

Number:

Invalid entry label to be displayed somewhere here

Email:

Gender: ☐ Male ☒ Female Birthdate:

dd

mm

yyyy

Add client

Clear Contents

Back

Edit a client

ID: {to be entered by user} Search

Name: {automatically generated when search button pressed and if entry is valid}

Number: {automatically generated when search button pressed and if entry is valid} Invalid entry label to be displayed somewhere here

Email: {automatically generated when search button pressed and if entry is valid}

Gender: ☐ Male ☒ Female Birthdate:

dd

mm

yyyy

Edit client

Clear Contents

Back

Add a new product

Company

Product name

Size

Rate

Product description

Add product

Clear contents

Back

Edit/Delete a product

Serial number Search

Company

Product name

Size

Invalid entry to be displayed somewhere here

Rate

Product description

Edit/Delete

Clear contents

Back

View Product Information_

Shows all product data of the product selected in the table

Product serial no.

Product company

Product name

Product description

Product size

Product rate

List of Products

View List

View Product Information

Back

List of products

ProductID	Product Company	Product name	Product description	Product Rate	Product Size

Button only visible when a row in the table is selected

3. Service Menu

Services

SERVICES

Add a new service

View services

Edit a service

Remove a service

Back

Add a service

Serial number

Category

Subcategory

Rate

Invalid entry to be displayed somewhere here

Add service

Clear contents

Back

Edit/Delete a service

Serial number

Category

Subcategory

Rate

Invalid entry to be displayed somewhere here

Edit/Delete

Clear contents

Back

List of services

View List

List of services

Back

Service ID	Category	Subcategory	Rate

4. Invoice Menu

Invoice

INVOICES

Prepare an invoice

View invoices

Back

Prepare an invoice

Prepare an invoice

Branch info

Select a branch

+

Client contact

To be entered

+

Product bought

Select a product

Qty

1

+

Service Taken

Select a service

+

Invoice date {automatically generated}

Client ID {automatically generated}

Client name {automatically generated}

Delete row

Item type	Description	Quantity	Rate	Amount

Generate invoice

Back

[illegible]

5. Branches Menu

Branches

BRANCHES

Add a new branch

View branches

Edit a branch

Delete a branch

Back

The image displays three screenshots of a branch management application, each showing a different form for managing branches.

Add a branch

This form is used to add a new branch. It contains the following fields and controls:

- BRANCH ID:** A text input field with the placeholder text "{automatically generated}".
- Address:** A text input field.
- Contact number:** A text input field.
- Invalid entry to be displayed somewhere here:** A label indicating where an error message should be shown.
- Buttons:** "Add branch", "Clear contents", and "Back".

Edit a branch

This form is used to edit an existing branch. It contains the following fields and controls:

- BRANCH ID:** A text input field with the placeholder text "{to be entered}".
- Address:** A text input field.
- Contact number:** A text input field.
- Invalid entry to be displayed somewhere here:** A label indicating where an error message should be shown.
- Buttons:** "Edit branch", "Clear contents", and "Back".

Delete a branch

This form is used to delete an existing branch. It contains the following fields and controls:

- BRANCH ID:** A text input field with the placeholder text "{to be entered}".
- Buttons:** "Delete branch", "Clear contents", and "Back".

[illegible]

6. Reports Menu

Reports

Generate Reports

Generate Report

Back

Login

Username

Password

Login

Generate Report

GENERATE REPORT

From {date 1} To {date 2} Generate

Amount of Sales

Number of Clients

Product Name	Product Frequency
Product1	Product1 Frequency
Product2	Product2 Frequency
Product3	Product3 Frequency
Product4	Product4 Frequency
Product5	Product5 Frequency

Service Name	Service Frequency
Service1	Service1 Frequency
Service2	Service2 Frequency
Service3	Service3 Frequency
Service4	Service4 Frequency
Service5	Service5 Frequency

Generate a PDF

Back

➤ Test Plan

Serial No.	Test Type	Nature of test
1	Upon starting the program, the main menu should come up with 5 sub menus and should be fully functional	Check if the menus and the toggle buttons are being displayed correctly and can navigate successfully and quickly through the different menus
2	Main menu should be hidden once a sub menu is opened	Check if the menu is being hidden after a sub menu is opened
3	Users should easily be able to navigate through the program	Check if all the buttons work properly
4	All database tables should be displayed to tables	Check if the View List button, which uses the SELECT query to retrieve data, works. To display this data to a JTable, use rs2xml.jar
5	Any new entries made should be added to the database and should be updated in the tables	Add new data and check if the data has been added to the tables by navigating through the program (example: add a client then check if that client has been added to the table or not)
6	Any deletion of data should successfully be removed from the database and the tables	Example: Firstly, check if the client exists, if exists, delete that client, then check the table and look for that client. If that client can't be found, then the client has been removed successfully
7	Manager should be able to successfully edit existing data and should be updated in the database and the tables	Firstly, check if the client exists, then edit client data using the "UPDATE" query and then check if the data has been updated in the client tables
8	Manager should be able to add any products or services to the list of products/services bought by the customer. It should be easy to use	Select a product or select a service and then press on the plus button to add that item to the list. Check if it has been added to the list or not
9	IDs and Serial numbers for the data should be unique and shouldn't be repeated	Using a unique way for making client and branch ID. For client ID, CLIENT + (number of seconds since 1970). This will always be different as the number of seconds since 1970 will continue to increase and will never be the same. For product, service, and invoice serial number, the auto increment function of MySQL was used, and it will be unique as whenever an entry is made, the previous value increases by 1, hence, preventing duplicate primary key values
10	Any invalid input should not be allowed into the database	For example: enter an alphabet or any other character other than numbers, an invalid entry message will be displayed and the value will not be allowed
11	PDF of the invoice should automatically be generated from the program. Managers should be able to open any invoice through the software	Prepare an invoice, and then save it at the desired location. Check that location in the computer and check if it exists. To check if it can be opened through the software, open the list of invoices, select

		an invoice row, and then press the open invoice button. Check if the correct invoice is opened or not
12	PDF of the report should automatically be generated from the program	First, choose 2 dates between which the report is to be generated. Then, click the generate pdf button to generate the pdf and save it to the desired location
13	Login page to access the reports menu	Unless the correct username and password are entered, the user will not be able to access the reports menu. The username and password will be provided to the owner