

Criterion B: Design

The designs and tables have been created using the client's requirements (**refer to Appendix Section A.2**)

Database Structure

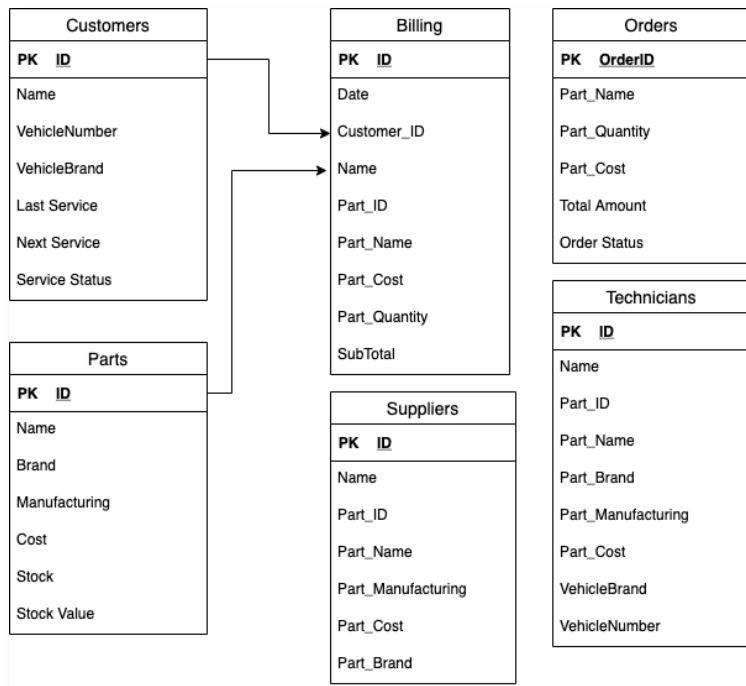


Figure 1: Entity Relationship Diagram

Proposed Tables-

Parts			
Field Name	Data Type	Validation	Example
Part_ID	VarChar	Cannot be null Must be unique	2346
Part_Name	VarChar	Must not contain numbers Can't be more than 10	Axel
Part_Brand	VarChar	Must not contain numbers Can't be more than 20	Amaron
Part_Manufacture	Date	Must only be of this format YYYY-MM-DD	2019-12-22
Part_Cost	Integer	All int values should be in Indian rupees	400
Part_Stock	Integer	All int values should be in Indian rupees	100
Total Stock Value	Integer	All int values should be in Indian rupees	40000

Suppliers			
Field Name	Data Type	Validation	Example
Supplier_ID	VarChar	Cannot be null Must be unique	1255
Supplier_Name	VarChar	Must not contain numbers Can't be more than 20	Vinod
Part_ID	Integer	Cannot be null Must not contain alphabets Must be unique	2346
Part_Name	VarChar	Must not contain numbers Can't be more than 10	Axel
Part_Brand	VarChar	Must not contain numbers Can't be more than 20	Amaron
Part_Manufacture	Date	Must only be of this format YYYY-MM-DD	2019-12-22
Part_Cost	Integer	All int values should be in Indian rupees	400

Customers			
Field Name	Data Type	Validation	Example
Customer_ID	VarChar	Cannot be null Must be unique	1354
Customer_Name	VarChar	Must not contain numbers Can't be more than 20	Vinod
Vehicle_Brand	VarChar	Must not contain numbers Can't be more than 20	Toyota
Vehicle_Number	VarChar	Must not contain numbers Can't be more than 10	HR26BY6612
Last_Serviced	Date	Must only be of this format YYYY-MM-DD	2018-01-02
Next_Service	Date	Must only be of this format YYYY-MM-DD	2020-05-22
ServiceStatus	VarChar	Shouldn't be empty	Complete

Technicians			
Field Name	Data Type	Validation	Example
Tech_ID	Integer	Cannot be null Must not contain alphabets Must be unique	2233
Tech_Name	VarChar	Must not contain numbers Can't be more than 20	Arjun

Technicians			
Part_ID	Integer	Cannot be null Must not contain alphabets	2346
Part_Name	VarChar	Must not contain numbers Can't be more than 10	Axel
Part_Brand	VarChar	Must not contain numbers Can't be more than 20	Amaron
Part_Manufacture	Date	Must only be of this format YYYY-MM-DD	2019-12-22
Part_Cost	Integer	All integer values should be in Indian rupees	400
Vehicle_Brand	VarChar	Must not contain numbers Can't be more than 20	Toyota
Vehicle_Number	VarChar	Must not contain numbers Can't be more than 10	HR26BY6612

UserLoginDetails			
Field Name	Data Type	Validation	Example
ID Number	VarChar	Cannot be null	2346
Username	VarChar	Must be unique	AS124
Password	VarChar	Cannot be null	Abc123
Access	Boolean	Can only be 1 or 0	—

Orders			
Field Name	Data Type	Validation	Example
OrderID	VarChar	Must be Unique	2346
Part_Name	VarChar	Should not be contain numbers	AS124
Part_Quantity	Integer	Should not be contain alphabets	100
Part_Cost	Integer	All int values should be in Indian rupees	40
TotalAmount	Integer	Should not be contain alphabets	4000
OrderStatus	VarChar	Should not be empty	

Billing			
Field Name	Data Type	Validation	Example
ID	VarChar	Cannot be null Must not contain alphabets Must be unique	2346
Date	Date	Must only be of this format YYYY-MM-DD	2020-12-22
Customer_ID	VarChar	Foreign Key connected to Customers Table	—
Name	VarChar	Must not contain numbers Can't be more than 20	Amaron
Part_ID	VarChar	Foreign Key connected to Parts Table	—
Part_Name	VarChar	Must not contain numbers Can't be more than 20	Axel
Part_Cost	Integer	All int values should be in Indian rupees	400
Part_Quantity	Integer	Must not contain alphabets	100
Sub_Total	Integer	All int values should be in Indian rupees	40000

Product designs

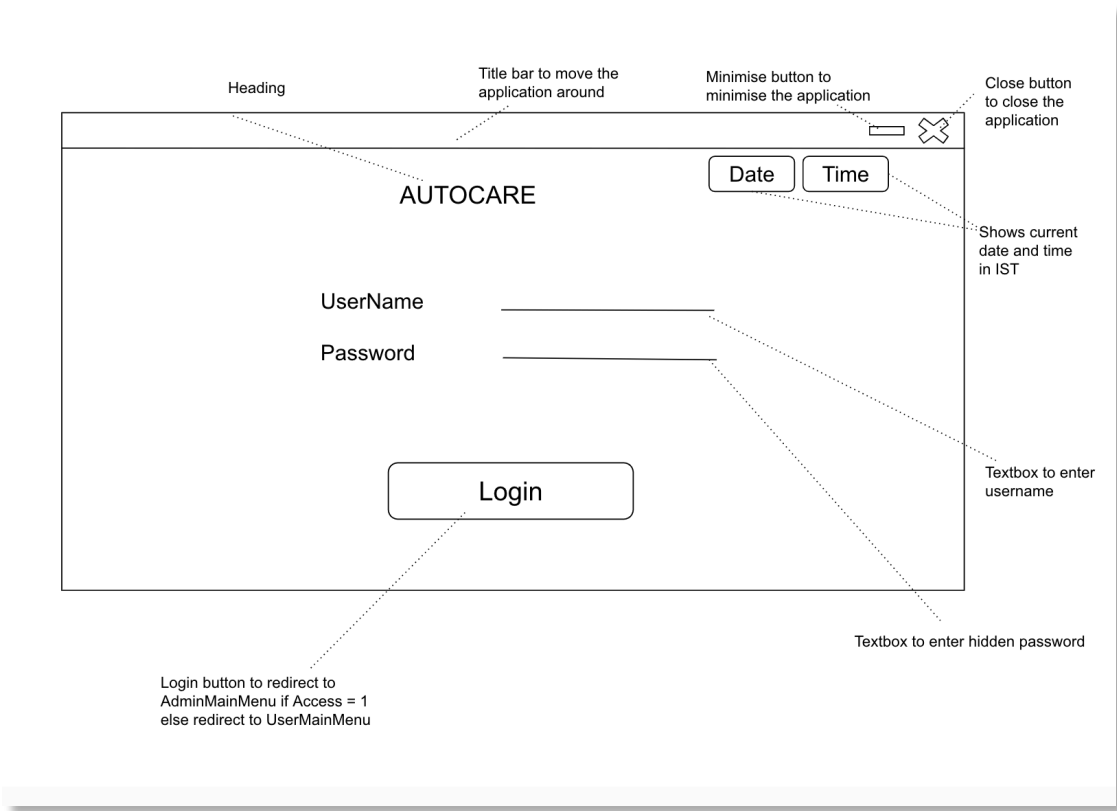


Figure 2: Login Form Design

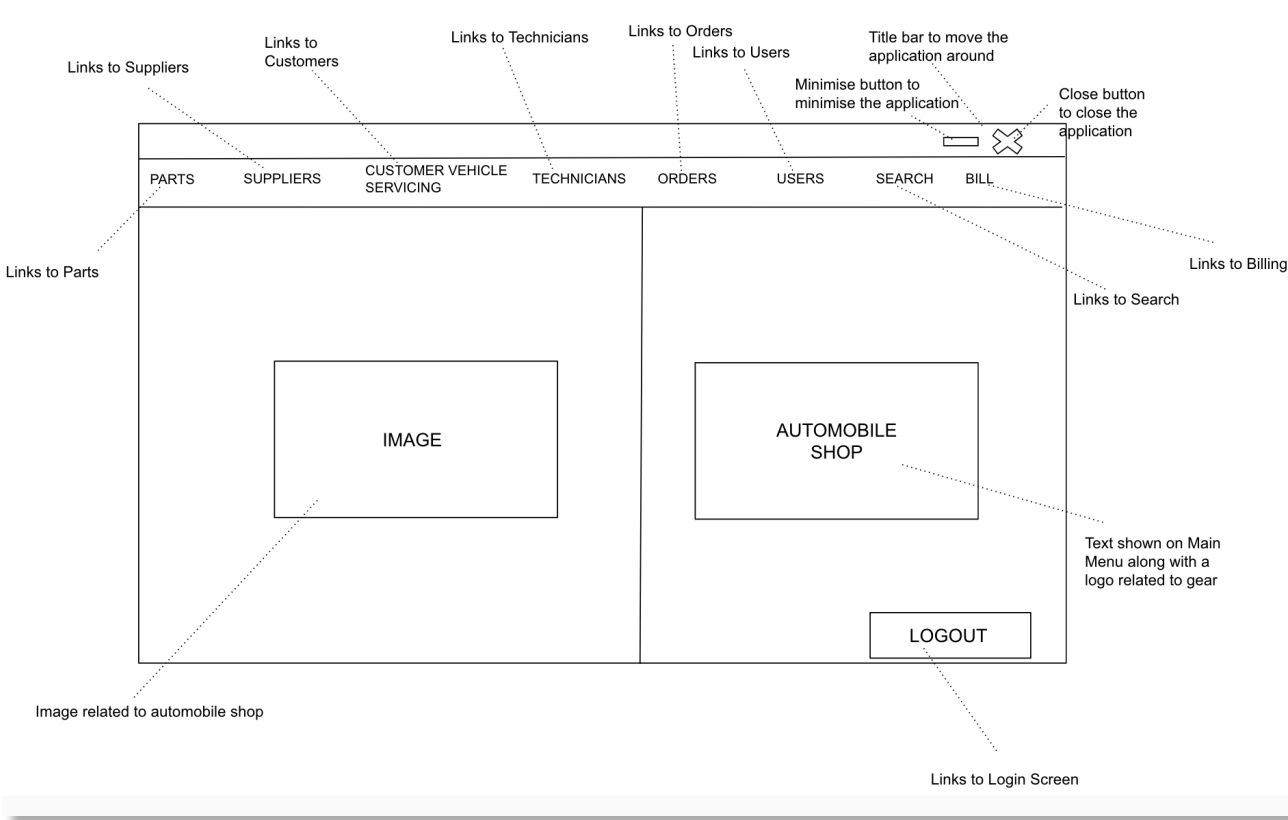


Figure 3: Admin Main Menu Design

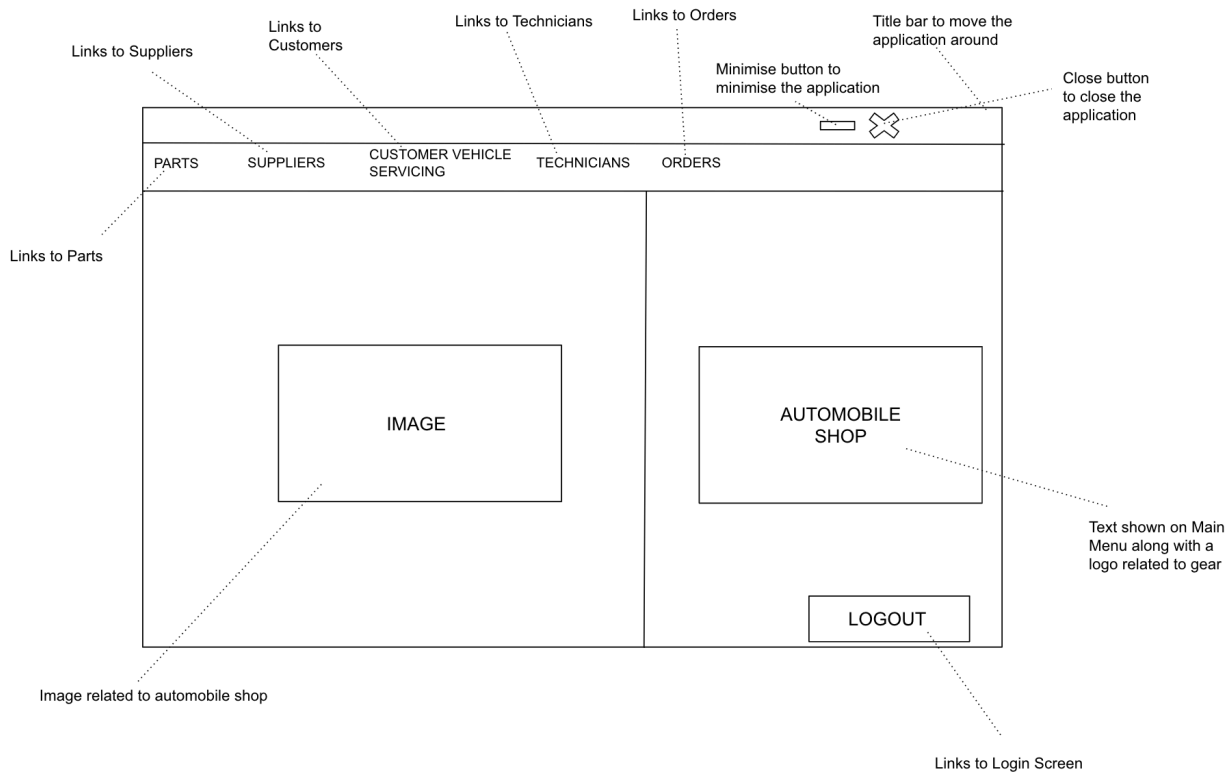


Figure 4: Employee Main Menu Design

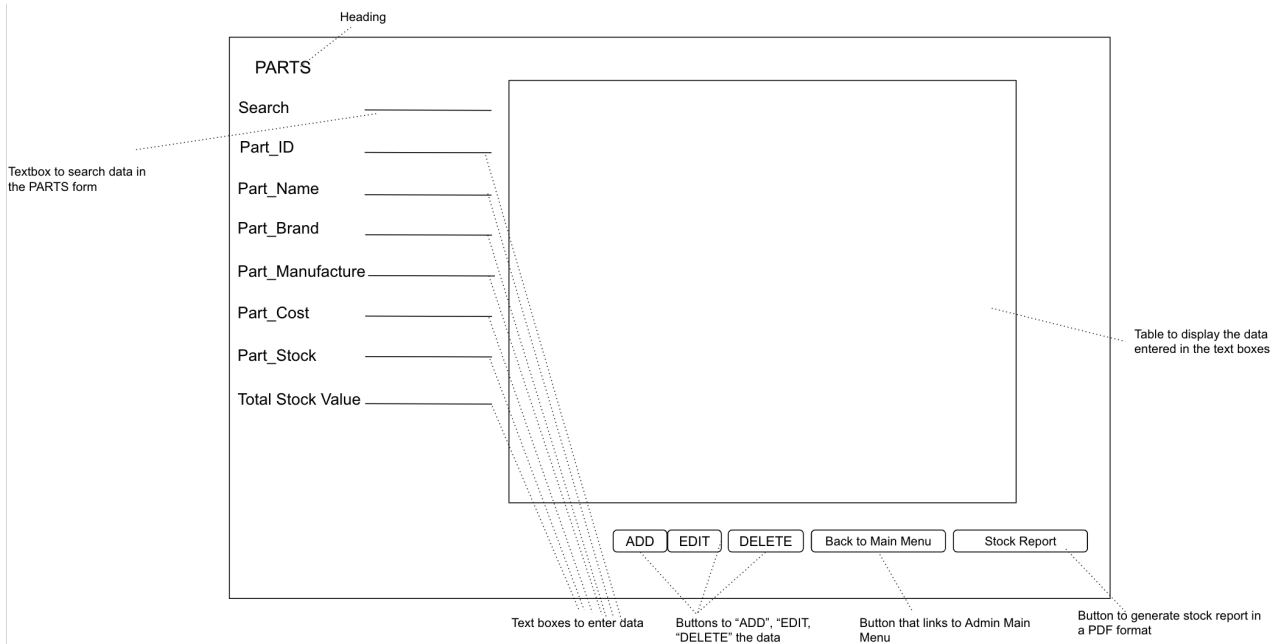


Figure 5: Parts Form Design

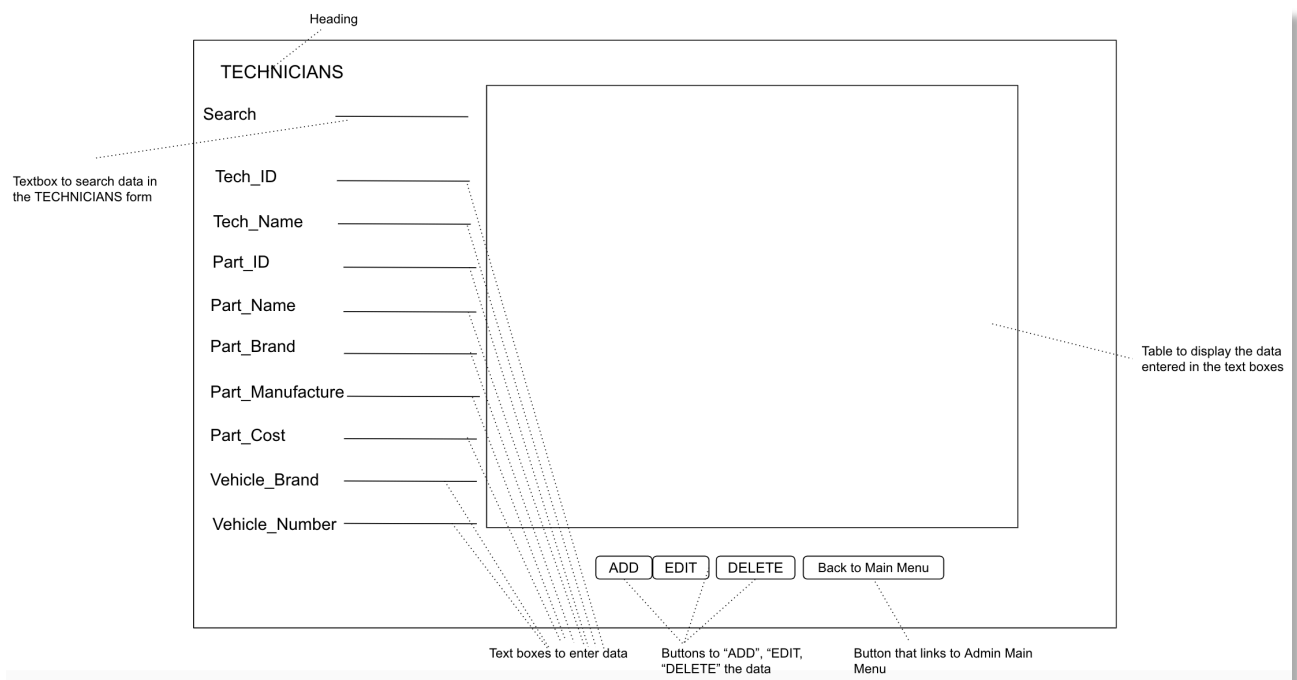


Figure 8: Technicians Form Design

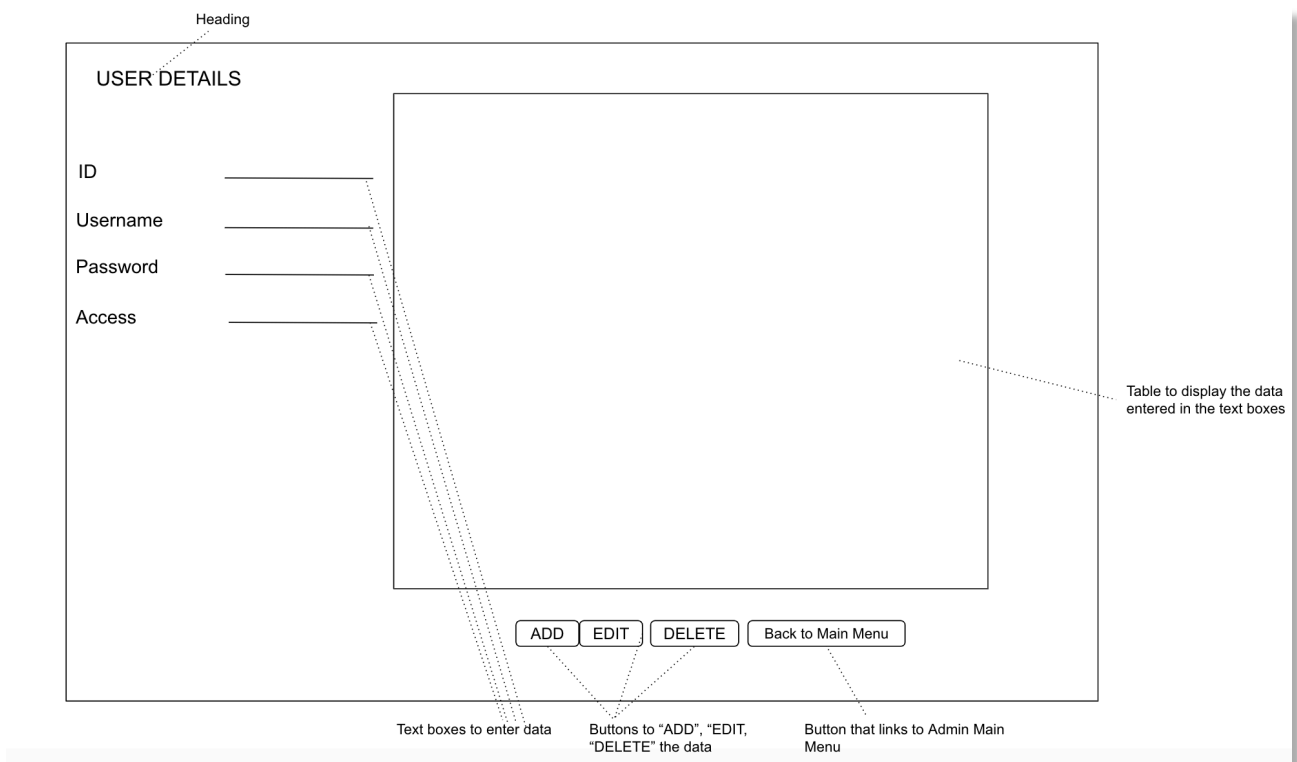


Figure 9: User Login Details Form Design

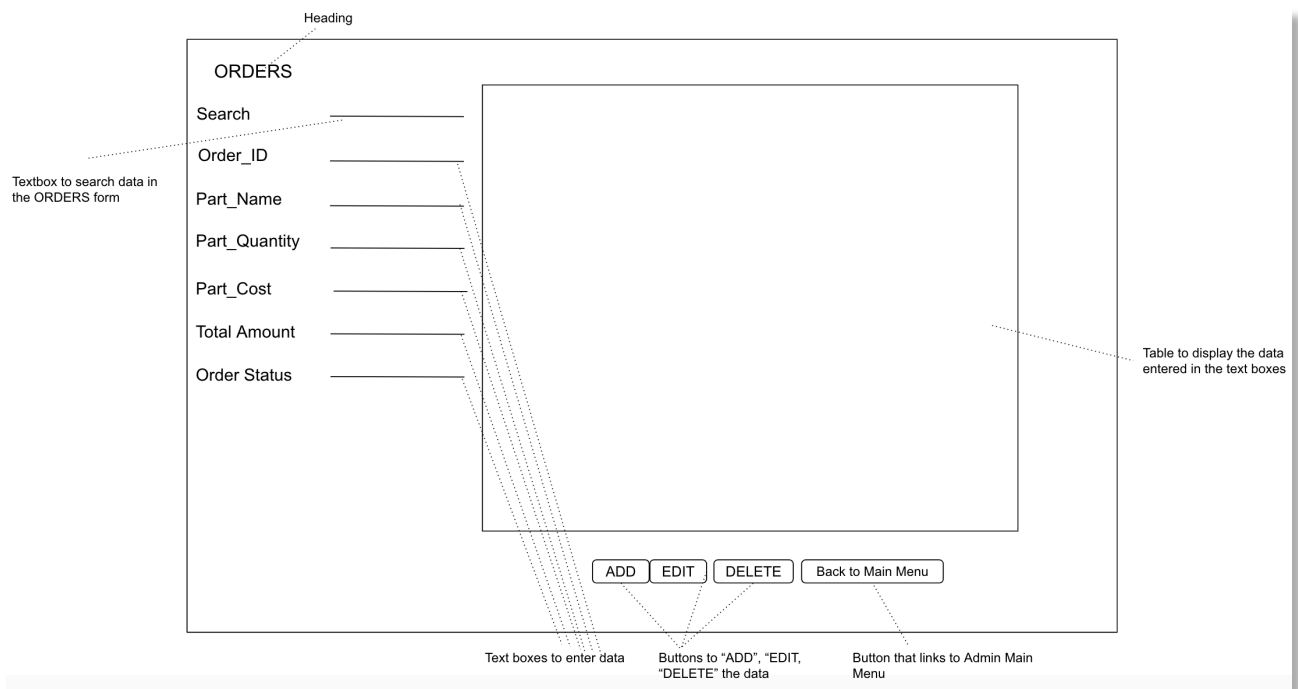


Figure 10: Orders Form Design

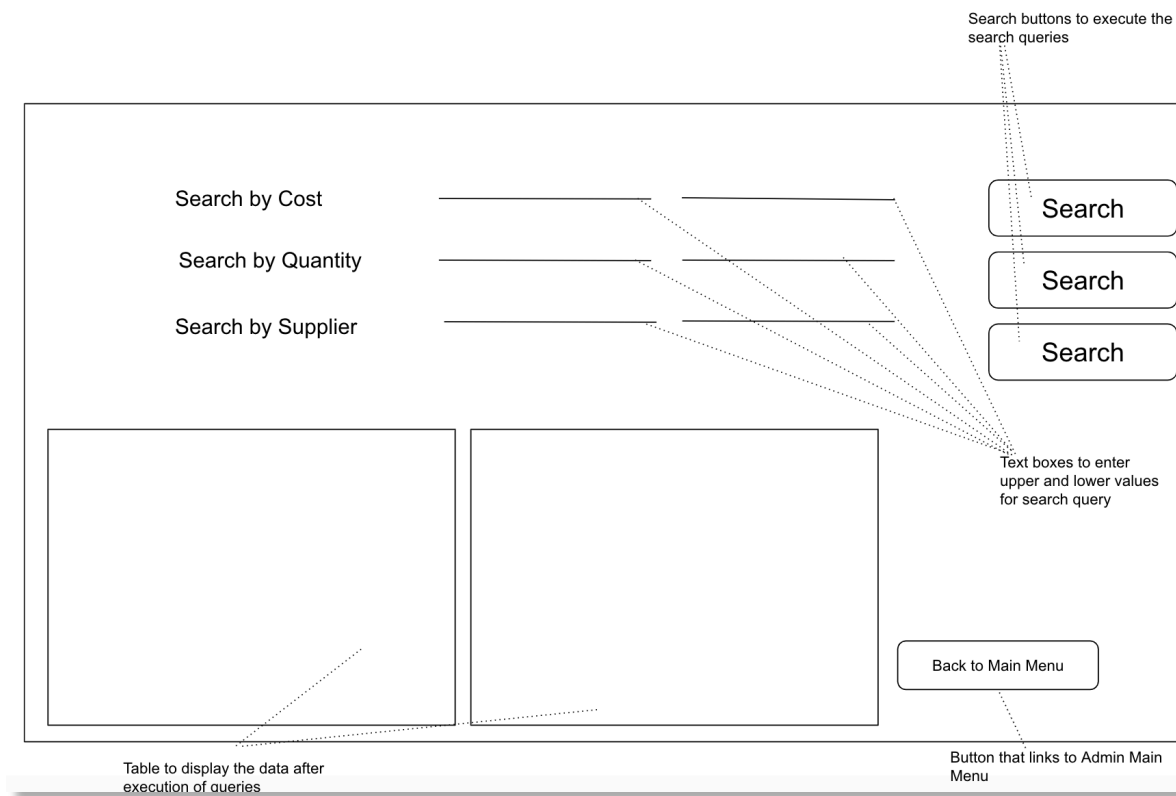


Figure 11: Search Form Design

The diagram illustrates a billing form layout. On the left, there is a vertical list of input fields: Date, Bill_ID, Customer_ID, Customer_Name, Vehicle_Brand, Vehicle_Number, Part_ID, Part_Name, Part_Cost, Part_Quantity, Total, Paid Amount, and Return Amount. Each field has a corresponding horizontal line for text entry. To the right of these fields is a large rectangular box representing a table for displaying entered data. Below the input fields, there are three buttons: 'ADD', 'RESET', and 'Back to Main Menu'. Callout lines point from descriptive text to various elements: 'Get automatically calculated' points to the 'Total' field; 'Text boxes to enter data' points to the input fields; 'Buttons to "ADD", "EDIT", "DELETE" the data' points to the 'ADD' and 'RESET' buttons; 'Button that links to Admin Main Menu' points to the 'Back to Main Menu' button; and 'Table to display the data entered in the text boxes' points to the large table area.

Figure 12: Billing Form Design

The diagram shows the 'PARTS DATA' section of an employee interface. It features a table with the following columns: Part_ID, Part_Name, Part_Brand, Part_Manufacture, Part_Cost, Part_Stock, and Total Stock Value. The table has one header row and one empty data row. Above the table is a heading 'PARTS DATA'. Below the table is a button labeled 'Back to Main Menu'. Callout lines provide context: 'Heading' points to 'PARTS DATA'; 'Table showing all parts data which can't be edited by employees' points to the table; and 'Button that links to Employee Main Menu' points to the 'Back to Main Menu' button.

Part_ID	Part_Name	Part_Brand	Part_Manufacture	Part_Cost	Part_Stock	Total Stock Value

Figure 13: Parts Form Design - Employee Interface

Heading

SUPPLIERS DATA

Supplier_ID	Supplier_Name	Part_ID	Part_Name	Part_Brand	Part_Manufacture	Part_Cost

Table showing all parts data which can't be edited by employees

Back to Main Menu

Button that links to Employee Main Menu

Figure 14: Suppliers Form Design - Employee Interface

Heading

CUSTOMER VEHICLE SERVICING DATA

Customer_ID	Customer_Name	Vehicle_Brand	Vehicle_Number	Last Serviced	Next Service	Service Status

Table showing all parts data which can't be edited by employees

Back to Main Menu

Button that links to Employee Main Menu

Figure 15: Customer Vehicle Servicing Form Design - Employee Interface

Heading

TECHNICIANS DATA

Tech_ID	Tech_Name	Part_ID	Part_Name	Part_Brand	Part_Manufacture	Part_Cost	Vehicle_Brand	Vehicle_Number

Table showing all parts data which can't be edited by employees

Back to Main Menu

Button that links to Employee Main Menu

Figure 16: Technicians Form Design - Employee Interface

Heading

ORDERS DATA

Order_ID	Part_Name	Part_Quantity	Part_Cost	Total Amount	Order Status

Table showing all parts data which can't be edited by employees

Back to Main Menu

Button that links to Employee Main Menu

Figure 17: Orders Form Design - Employee Interface

Processing Flowcharts:

After finalising the functionalities required by client, I created flowcharts to understand the proper processes.

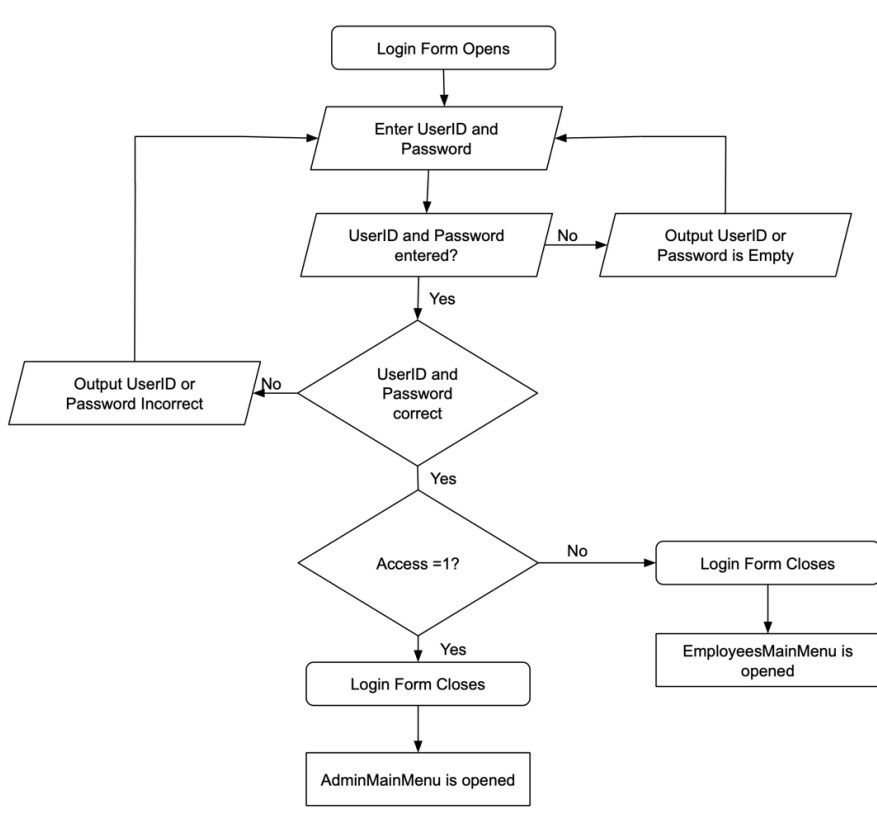


Figure 18: Login Form Flowchart

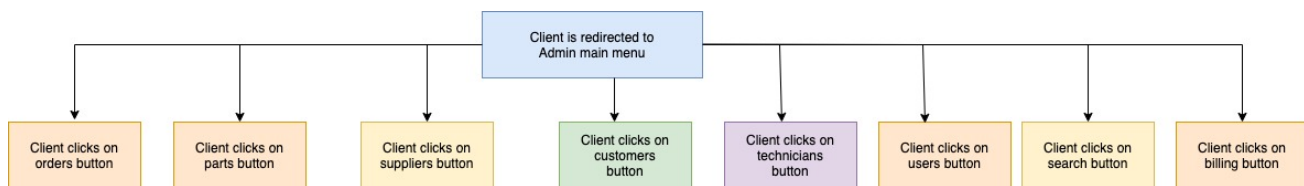


Figure 19: Admin Main Menu Flowchart

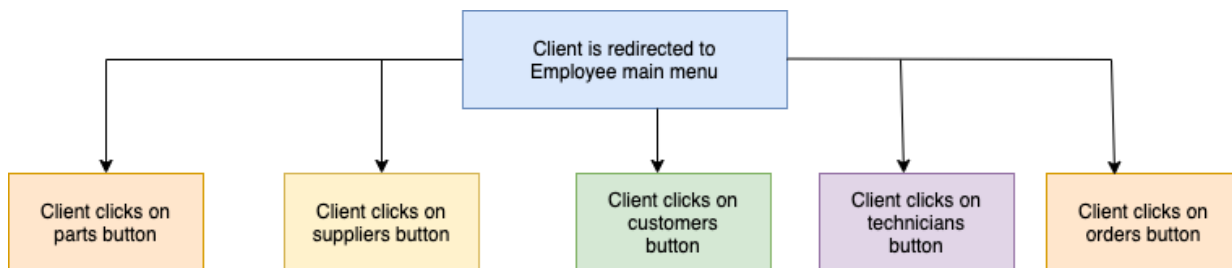


Figure 20: Employee Main Menu Flowchart

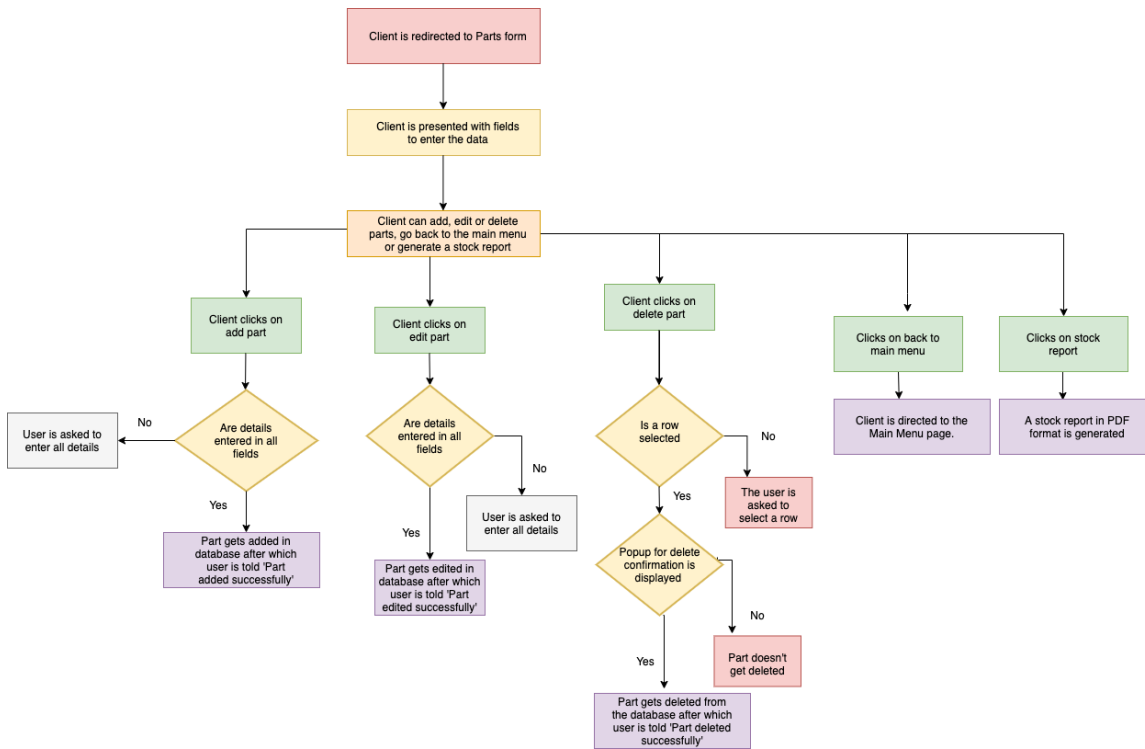


Figure 21: Parts Form Flowchart - Admin Interface

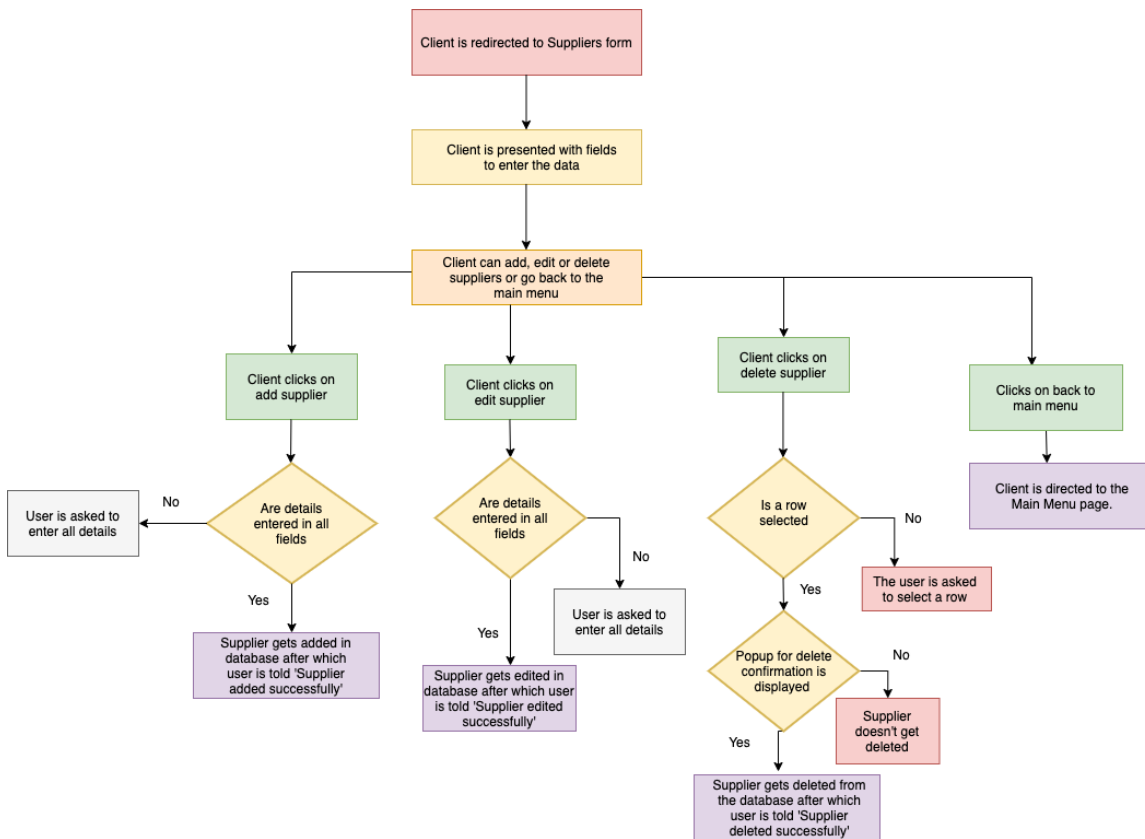


Figure 22: Suppliers Form Flowchart - Admin Interface

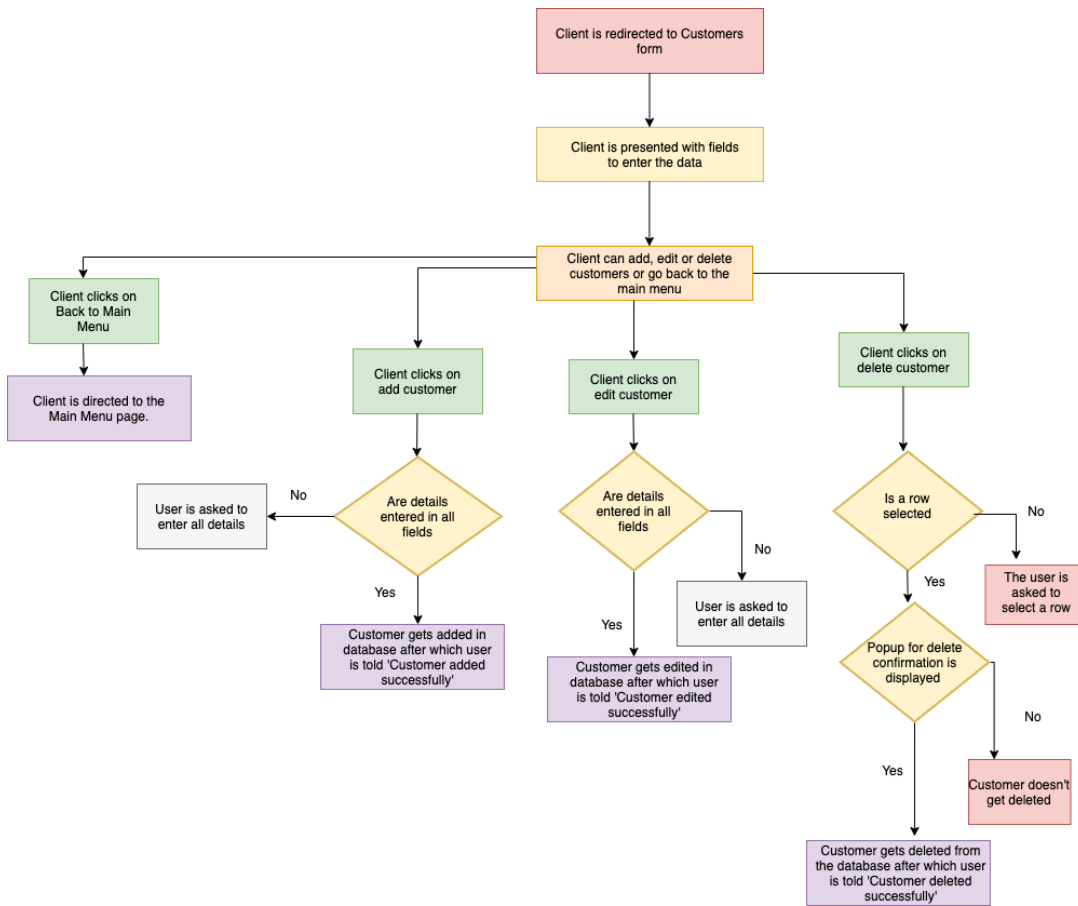


Figure 23: Customer Vehicle Servicing Form Flowchart - Admin Interface

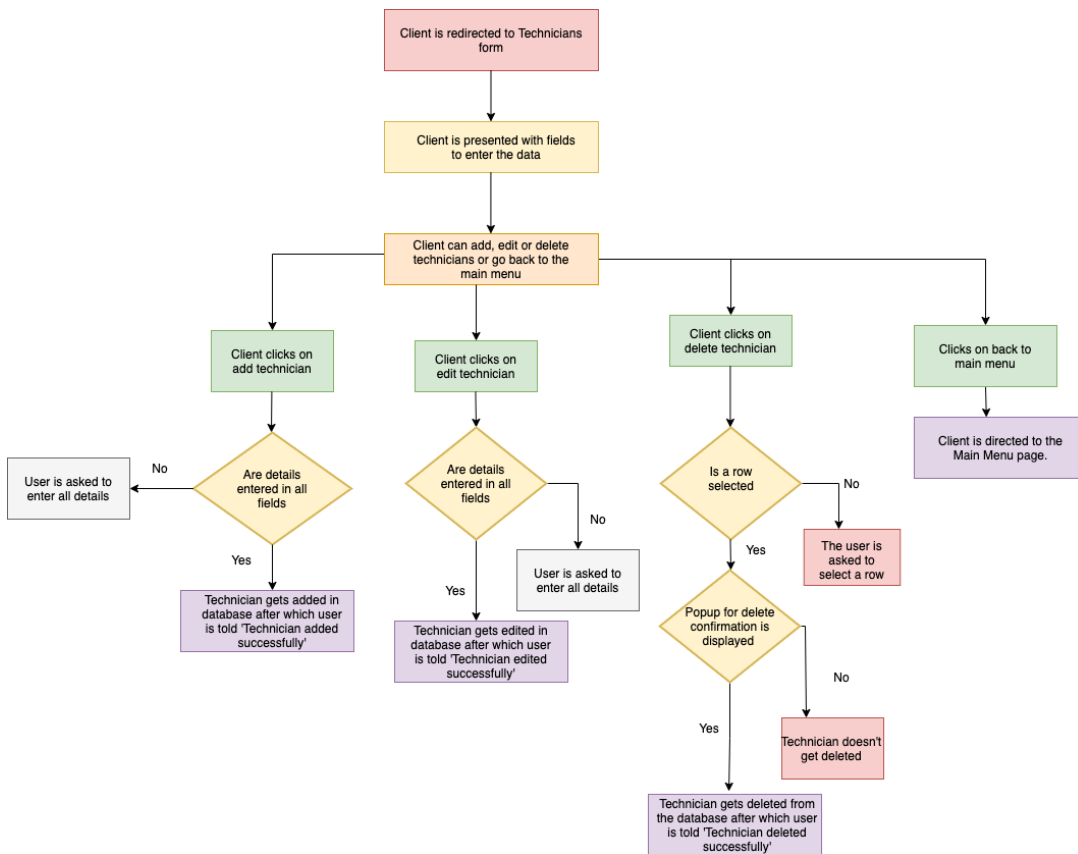


Figure 24: Technicians Form Flowchart - Admin Interface

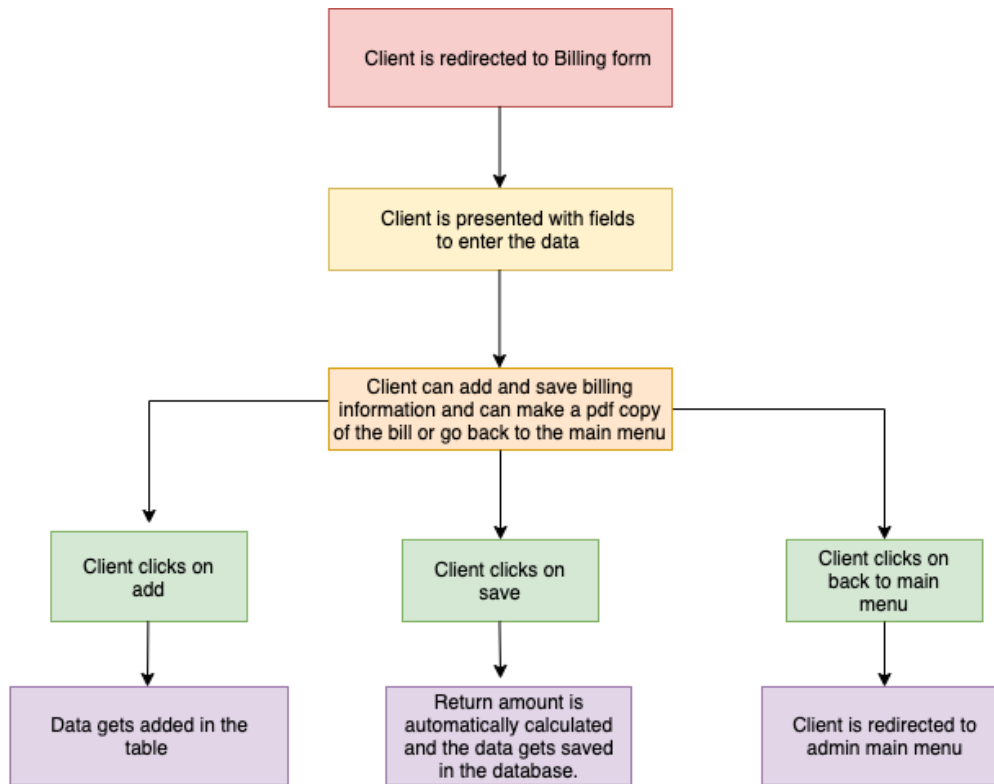


Figure 25: Billing Form Flowchart - Admin Interface

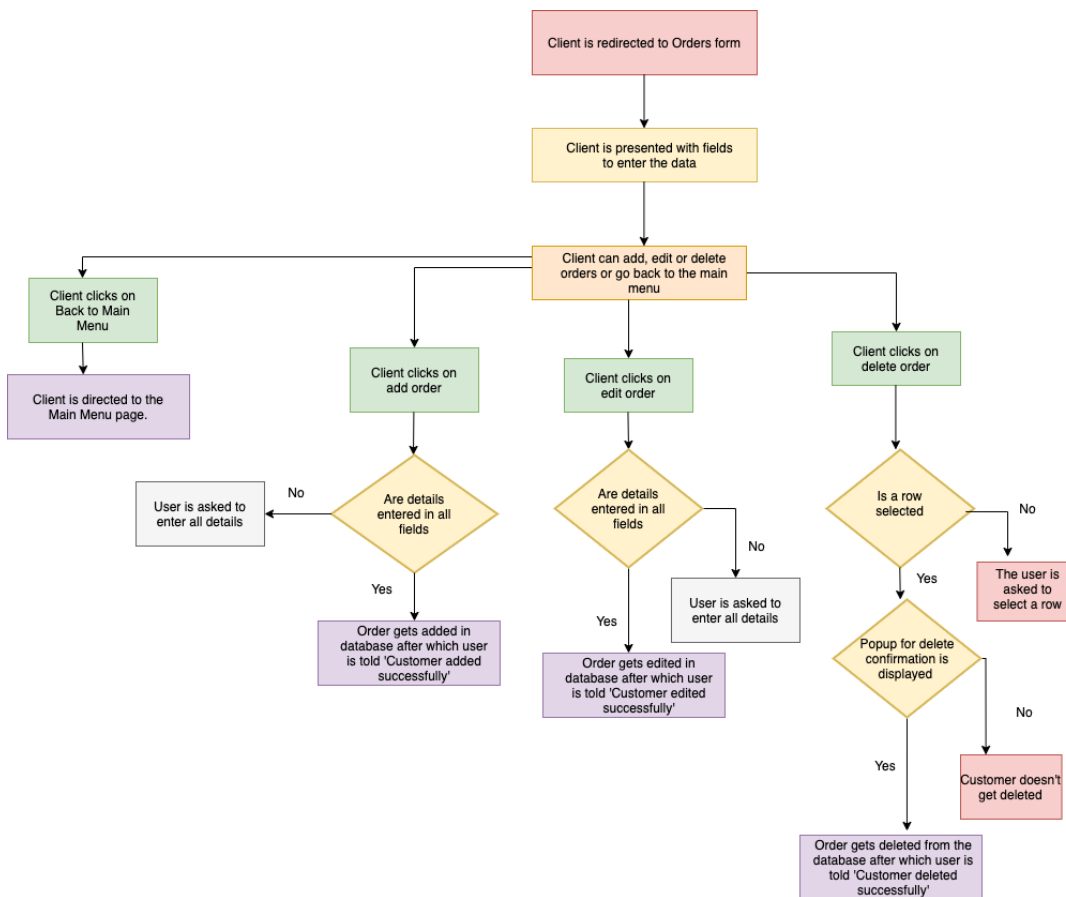


Figure 26: Orders Form Flowchart - Admin Interface

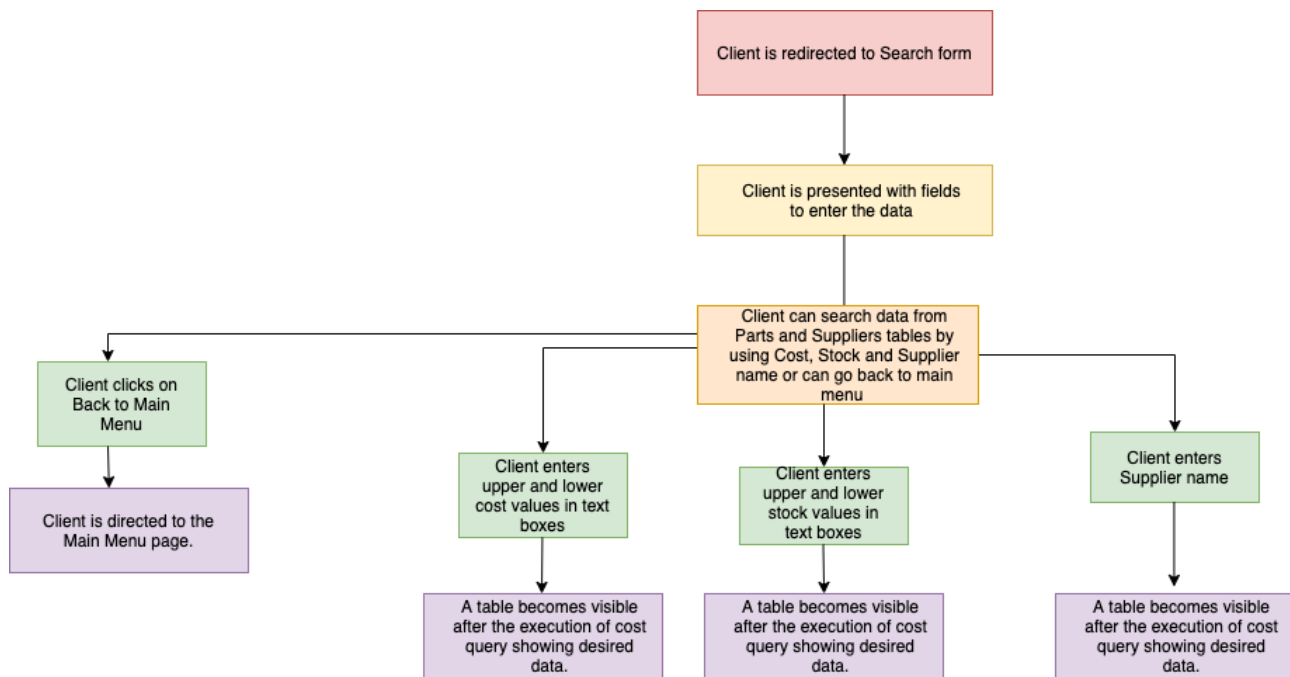


Figure 27: Search Form Flowchart - Admin Interface

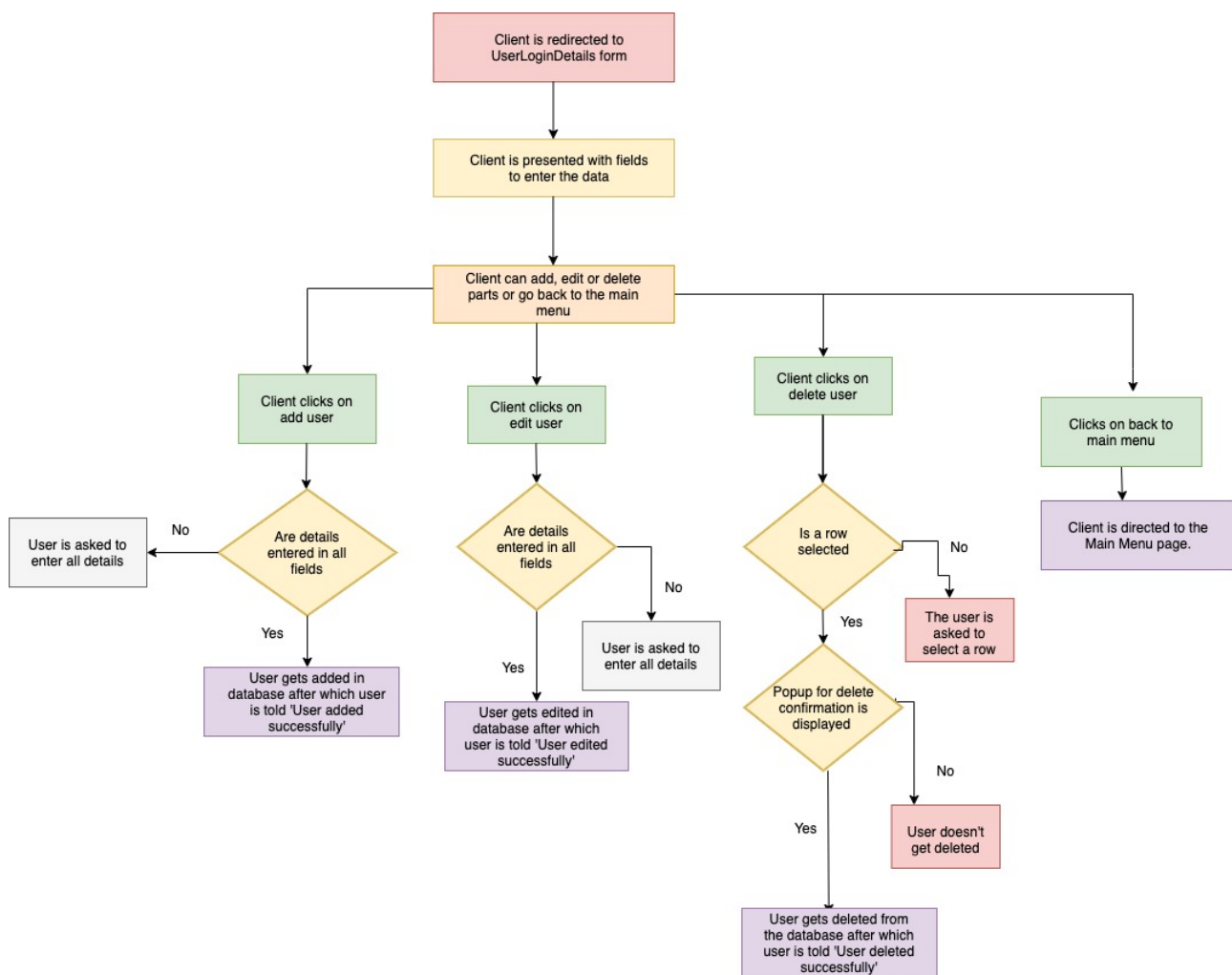


Figure 28: User Login Details Form Flowchart - Admin Interface

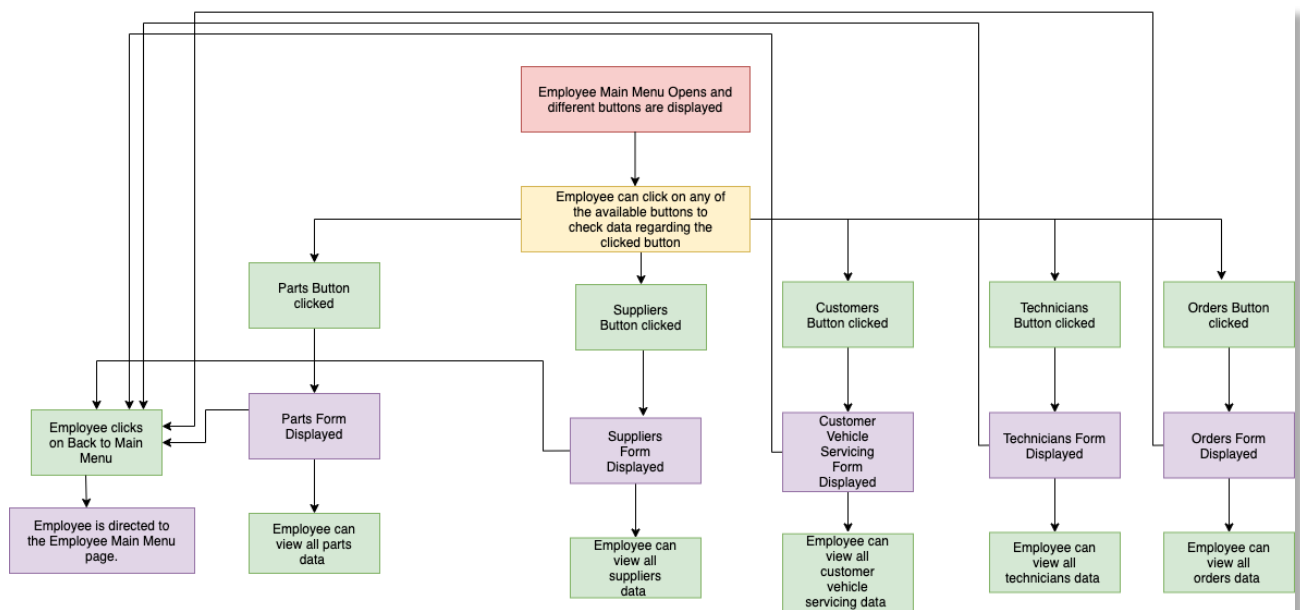


Figure 29: Employee Interface Flowchart

UML Diagram:

I have created a UML Diagram to help me understand the working of the classes and all of its variables and methods that will allow me to create it in the actual product.



Figure 30: UML Diagram - Admin Interface



Figure 31: UML Diagram - Employee Interface

Development of Pseudocode:

When the client confirmed the process flow, I decided to prepare pseudocode before coding.

Database Connectivity:

```
Connection con = null //variable of type Connection
if con=null //Checking if the connection exists
then try
    setting the JDBC driver as the default driver
    accessing the location of the DB
    setting the JDBC connection to the database
    setting con equal to the connection of the database
    show 'Database Connected'
catch exception
    display error 'Database not connected'
end if
```

Login Interface:

```
if Username and Password text boxes are filled && Username and Password
match
then output "Login Successful" and open Admin Main Menu if 'Access' = 1 else
open Employee Main Menu
else if Username and Password text boxes are not filled
then display "please enter Username or Password"
else if Username and Password do not match
then display "UserID or Password Incorrect" end if
```

Main menu - Admin Interface:

```
If PartsButton = true
    then open Parts Form
else if SuppliersButton = true
    then open Suppliers Form
else if CustomersButton = true
    then open Customers Form
else if TechniciansButton = true
    then open Technicians Form
else if OrdersButton = true
    then open Orders Form
else if UsersButton = true
    then open UserDetails Form
end if
```

Main menu - User Interface:

```
If PartsButton = true
    then open Parts Form
else if SuppliersButton = true
    then open Suppliers Form
else if CustomersButton = true
    then open Customers Form
else if TechniciansButton = true
    then open Technicians Form
else if OrdersButton = true
    then open Orders Form
end if
```

Add Button

```
If all details are filled && AddButton = true
display 'data added successfully' and display the details of part added in the table
else display 'please enter all details' and display table as it is
```

Edit Button

```
if all details are filled && EditButton = true
Display 'data edited successfully' and display the edited details of part in the table
else display 'please enter all details' and else display table as it is
```

Delete Button

```
if DeleteButton = true
If rowselected = true
display 'are you sure you want to delete the record'
If Okbutton = true
then Display 'record deleted successfully' and display the details of the table apart
from record which is deleted
else display table as it is
else if Cancelbutton = true
Redirect to Parts Form
If rowselected = false
Display 'please select a row'
```

Back to Main Menu Button

```
If BacktoMainMenu button = true
Then show Main Menu
```

Generating Stock Report Button in Parts Form

If StockReport button = true
Generate and save a stock report in CS IA folder

Calculating total stock and total order amount value

Int x = PartCost
Int y = PartStock
Int z = x*y
Total Stock Value = z

Logout Button

If Logout button = true
Then open Login Form

Test Plan:

The test cases were verified by my CS teacher, a friend before handing over to my client. Finally the product was validated as per the Test plan by the client.

Test #	Actions to Test	Method of Testing & Expected Result
TC #1	Database connectivity with Java	Using the MySQL command prompt store random data in the database, This data should be stored in the database in a tabular format
TC #2	Verify that after opening the application, the User is prompted to enter their login information.	After opening the application, the User should be prompted with the Login screen with User ID and Password text boxes and Login button.
TC #3	Verify that desired Main Menu form is opened after selecting user type	If correct UserID and Password are entered, user is granted access into application. If incorrect UserID or Password are entered, user or admin is not granted access into application. If Access = 1, user is redirected to Admin Main Menu else is redirected to Employee Main Menu.
TC #4	Verify that when user clicks on any of the buttons in main menu button user is redirected to the clicked button's form only and not any other form	After clicking on PARTS user should be redirected to PARTS form only. After clicking on SUPPLIERS user should be redirected to SUPPLIERS form only. After clicking on CUSTOMERS user should be redirected to CUSTOMERS form only. After clicking on TECHNICIANS user should be redirected to TECHNICIANS form only. After clicking on ORDERS user should be redirected to ORDERS form only. After clicking on USERS user should be redirected to USERS form only. After clicking on BILL user should be redirected to BILL form only.
TC #5	Verify that 'Add' button functionality on all forms works properly	After clicking on 'Add' button in all forms the Client should be able to enter details related to the specific forms and after clicking on 'Add' button, the data should be saved in the database.

Test #	Actions to Test	Method of Testing & Expected Result
TC #6	Verify that 'Edit' button functionality on all forms works properly	After clicking on 'Edit' button in all forms the Client should be able to edit details related to the specific forms and after clicking on 'Edit' button, the data should be saved in the database.
TC #7	Verify that 'Delete' button functionality on all forms works properly	A) The Client should be able to select a record to delete and after pressing on 'Delete' button, the confirmation prompt should be displayed. After clicking 'Ok' button, the data should be deleted from database. B) The Client should be able to select the part to delete and after pressing on 'Delete' button, the confirmation prompt should be displayed. After clicking Cancel button, the data should not be deleted from database.
TC #8	Verify that 'Back to Main Menu' button works on all forms of Admin interface	Once clicking on 'Back to main menu' button the client should be redirected to the main menu form of the Admin interface.
TC #9	Verify that Search Bar works on all forms of Admin interface	The search results are shown in Jtable when something is written in the search bar present in all forms.
TC #10	Verify that Search Buttons in Search Form work.	When search query is executed using Part_Cost, the table shows the data according to Part_Cost query only. When search query is executed using Part_Stock, the table shows the data according to Part_Stock query only. When search query is executed using Supplier_Name, the table shows the data according to Supplier_Name query only.
TC #11	Verify that 'Logout' on both Main Menus works properly	After clicking on 'Logout' button, the User should be able to exit from the screen and redirected to Login Screen.
TC #12	Verify that Stock Report button works in the Parts Form	Upon clicking the "Stock Report" button, a PDF file containing all the data of the parts form should be generated.
TC #13	Verify that 'Add' button works in Billing Form	Upon clicking 'Add' button in Billing Form, data from both Parts and Customer database should be saved in the Billing Form database.
TC #14	Verify that data is displayed in fields in Billing Form after entering ID number.	Upon entering Customer ID and Part ID, the data having the same ID number should be displayed in all fields.
TC #15	Verify that Return Amount in Billing form is automatically calculated	Upon entering Total and Paid Amount, the return amount should automatically be calculated.
TC #17	Verify that 'Save' button works in Billing Form	After data getting saved in the database, once the save button is clicked in billing form, one particular record needs to be exported in a PDF format.
TC #18	Verify that ID number is automatically generated in the Billing Form	ID number should be automatically generated and incremented once data in the Billing form gets saved in the database.

Test #	Actions to Test	Method of Testing & Expected Result
TC #19	Validations	To make data adhere to data type and there is correct information in the database. Error box should output a warning message, notifying the user to input data in the correct format. Otherwise data should be recorded.