

**High Level Design & Low Level Design**

**Document Control :**

# Project Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **Version** | **Author** | **Brief Description of Changes** | **Approve**  **Sign** |
| 27.09.2022 | 1.0 | Group 3 |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**INDEX**

1. Introduction ------------------ 4

2. Purpose ------------------ 4

3. Project Scope ------------------ 4

4. Project Audience ------------------ 4

5. System Architecture ------------------ 4

6. Function Overview ------------------ 5

int show\_customer () ------------------ 5

int add\_product () ------------------ 5

int modify\_stock() ------------------ 5

int modify\_customer() ------------------ 6

int file\_maintainance () ------------------ 6

int customer\_registration() ------------------ 6

int consult\_order() ------------------ 6

int show\_product() ------------------ 6

int checkout () ------------------ 6

int new\_order () ------------------ 6

int show\_all\_orders() ------------------ 6

int show\_delivery() ------------------ 7

int begin\_delivery() ----------------- 7

7. Design Objective ----------------- 8

8. Design Alternative ----------------- 9

9. Error Detection / Exception Handling

10. Validations

11. Pseudocode ----------------- 9

12. UML Diagrams ----------------- 9

12.1 Sequence Diagram ----------------- 9

12.2 Class Diagram ----------------- 10

13. ER Diagram ------------------- 11

14. Data Flow Diagram ------------------- 12

* **Introduction**

The Oozy Online Snack Delivery Bar is a menu driven online shopping system that allows user to search through the snacks and buy maximum three types of snacks at a time. Program allows the customer to view snacks details and order them, and all these changes are stored and can be viewed anytime by the employee. Only an authorized company person is allowed to update customer and snacks database. Delivery of orders is based on descending order of distance entered by the customer that is the customer nearest to the snack bar will receive the order first.

* **Purpose**

Main objective of The Oozy Online Snack Delivery Bar is to provide users a better and convenient interface to order snacks online. This online portal can also be beneficial in bridging the gap between sellers and consumers. This is an user-friendly menu based system for both customer and employees.

* **Project Scope**

This project aims to create a user-friendly online snacks delivery system where user is allowed to buy maximum of three type of snacks at a time. It can generate a detailed sales invoice. All the information about a particular customer or a product is stored in a retrievable manner.

* **Project Audience**

The audience are the people who wants to buy snacks online using this online snack delivery system.

* **System Architecture**

**The Oozy Online Snack Delivery Bar**

This is a menu driven online snacks delivery system that allows user to search through the snacks and buy maximum three types of snacks at a time. In this online snack delivery system orders are delivered based on distance entered by the customer. Program allows the user to view snacks details and purchase item, and all these changes are stored and can be viewed anytime. Only an authorized company person is allowed to update customer and snacks database.

* **Functional Overview**: -

## Following header files are included in the program:

* #include <bits/stdc++.h>
* #include <fstream>
* #include <termios.h>
* #include <unistd.h>
* #include <stdio.h>
* #include <thread>
* #include <mutex>

**int show\_customer\_list();**

This function allows the employee to view the customer’s details i.e name, customer\_id, phone number,email\_id and custtomer id is generated automatically.

**int add\_product();**

This function allows the employee to add product name,product description,units available,price and product id is generated automatically.

**int modify\_product();**

This function takes product id as input and ask the employee to modify stock details that is name,description,units available,unit price and also have an option to delete the product.

**int modify\_customer();**

This function takes customer id as input and ask the employee to change the customer details that is name,phone no,email and also have an option to delete customer.

**int file\_maintenance();**

This is password authenticated function accesible only for employees having a sub menu to add product,modify stock,modift customer and to view customer list.

**int customer\_registration();**

This function allows the customer to register themself to the oozy snack online delivery system by adding details like name,email,phone no and password.

**int consult\_order();**

This function shows the details of the order after entering order id.

**int show\_products();**

This function shows all the available snacks list to the customer and employee in the snack bar menu.

**int checkout();**

This function allows customer to checkout after the snack purchase of maximum three items.

**int new\_order();**

In this function customer has to enter phone no and password and after validation they are allowed to place an order of maximum three snacks .

Customer also has to enter the distances between snack bar and their delivery location.

**int show\_all\_orders();**

This function displays all the orders placed by the customer in an organised format.

**int show\_delivery();**

This function shows the delivery list in descending order of the distance.

**int begins\_delivery();**

This function begins the delivery in descending order of the distance.

The Oozy Online Snack Delivery System comprises of the following modules:

|  |  |
| --- | --- |
| Name of the Module | Customer Registration |
| Handled by | Ojas Bhimta |
| Description | Customer must register for the first time to avail benefits of the Online Snack Delivery System |

|  |  |
| --- | --- |
| Name of the Module | New Order |
| Handled by | Ojas Bhimta |
| Description | To place the order of maximum 3 snacks at the time |

|  |  |
| --- | --- |
| Name of the Module | Show all order |
| Handled by | Samiksha Parakh |
| Description | Shows details of all the orders. |

|  |  |
| --- | --- |
| Name of the Module | Product Availability List |
| Handled by | Jaanvi Gujral |
| Description | Shows details of all the snacks available to the user . |

|  |  |
| --- | --- |
| Name of the Module | Consult order |
| Handled by | Apurva |
| Description | Shows details of specific order after user enters the order id. |

|  |  |
| --- | --- |
| Name of the Module | Show delivery list |
| Handled by | Apurva |
| Description | Shows details of all the pending orders to the employees. |

|  |  |
| --- | --- |
| Name of the Module | Begin Delivery |
| Handled by | Ojas Bhimta |
| Description | Menu based function to begin delivery of orders |

|  |  |
| --- | --- |
| Name of the Module | File Maintenance |
| Handled by | Jaanvi Gujral |
| Description | Menu based function to add product, modify product, modify customer and view customer. |

|  |  |
| --- | --- |
| Name of the Module | Show order by min value |
| Handled by | Samiksha Parakh |
| Description | Display details of orders having total payable amount more than min value. |

### Design Objectives: -

* Customer Registration
* New Order
* Show all Orders
* Product Availability List
* Consult Order
* Show Delivery List
* Begin Delivery
* File Maintenance
* Show order by min value
* Exit

**Design Alternative: -** we have used mostly the C++ concepts to perform all the relevant operations in the particular files .

### Error Detection / Exceptional Handling: -

User should first enter the details according the condition and if the entered detail is not according the condition specified sometimes it is displays the message that is entered and sometimes it returns with an error.

**Validation:-**

* Customer name should be minimum of three and maximum of ten characters.
* PhoneNo should not be greater than 10 digits.
* PhoneNo should not start with 0.
* It can not contain any alphabet or special characters.
* Product name should be minimum of 3 or maximum of 10 characters.
* Unit Price for the snacks must be greater than zero.
* Unit Price must be a numeric value.
* Unit available for the snacks cannot zero.
* They should be always of integer type.
* Description for the product should always be greater than three and less than ten characters.
* Space and special characters are not allowed in any of the main menu function

**PseudoCode**

begin

int flag=0;

string phone

display "enter phone number"

accept phone

for it=cust.begin() to cust.end

if(it->get\_phone()==phone)

flag=1

display enter password

string pass

accept pass

if(it->get\_pass!=pass)

display "invalid password"

return

end if

int choice

while (choice!=3)

display "1.continue shopping"

display "2.checkout"

display "3.cancel and proceed to main menu"

accept choice

if(ch==1)

if(vord.size()==3)

display "item limit exceeded.you can either checkout or cancel or proceed to main menu"

continue

end if

int flag=0

long long int pid

display "enter product id"

accept product id

for pit=product.begin() to product.end

if(pit->get\_id==pit)

pflag=1

int req

display "enter units required"

accept req

if(pit->get\_ua() >= req)

order tmp(pid,pit->get\_ua(),pit->get\_pname(),req)

vord.push\_back(tmp)

display "item added to cart"

end if

display "entered quantity not available"

end if

end for

if(!flag)

display "product not found"

end if

else if(ch==2)

float d

while(1)

display "enter distance between snacks bar and your delivery location"

accept d

if(d<=0)

display "invaliid input"

end if

else break

end while

final\_order tmpord(it->get\_cid),phone,it->get\_cname()

ord.push\_back(tmpord)

checkout()

tmpord.show\_final\_order()

return 0

end if

else if(ch==3)

return 0

end if

else

display "invaliid input"

end if

end for

if(!flag)

display "phone number not registered.please register through customer registration option in main menu"

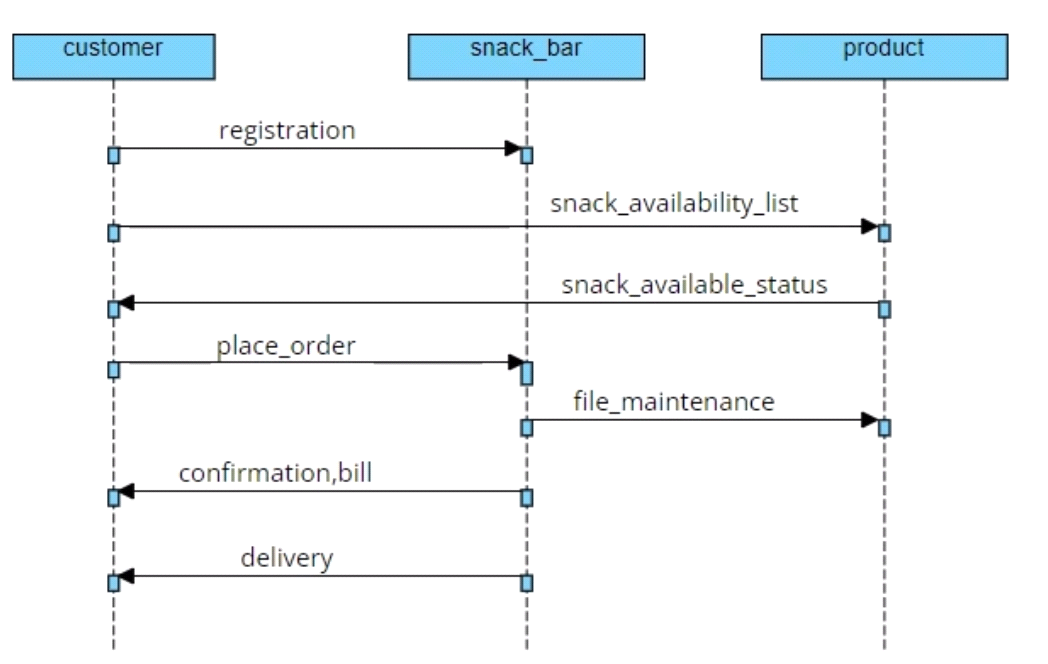
return 0

end if

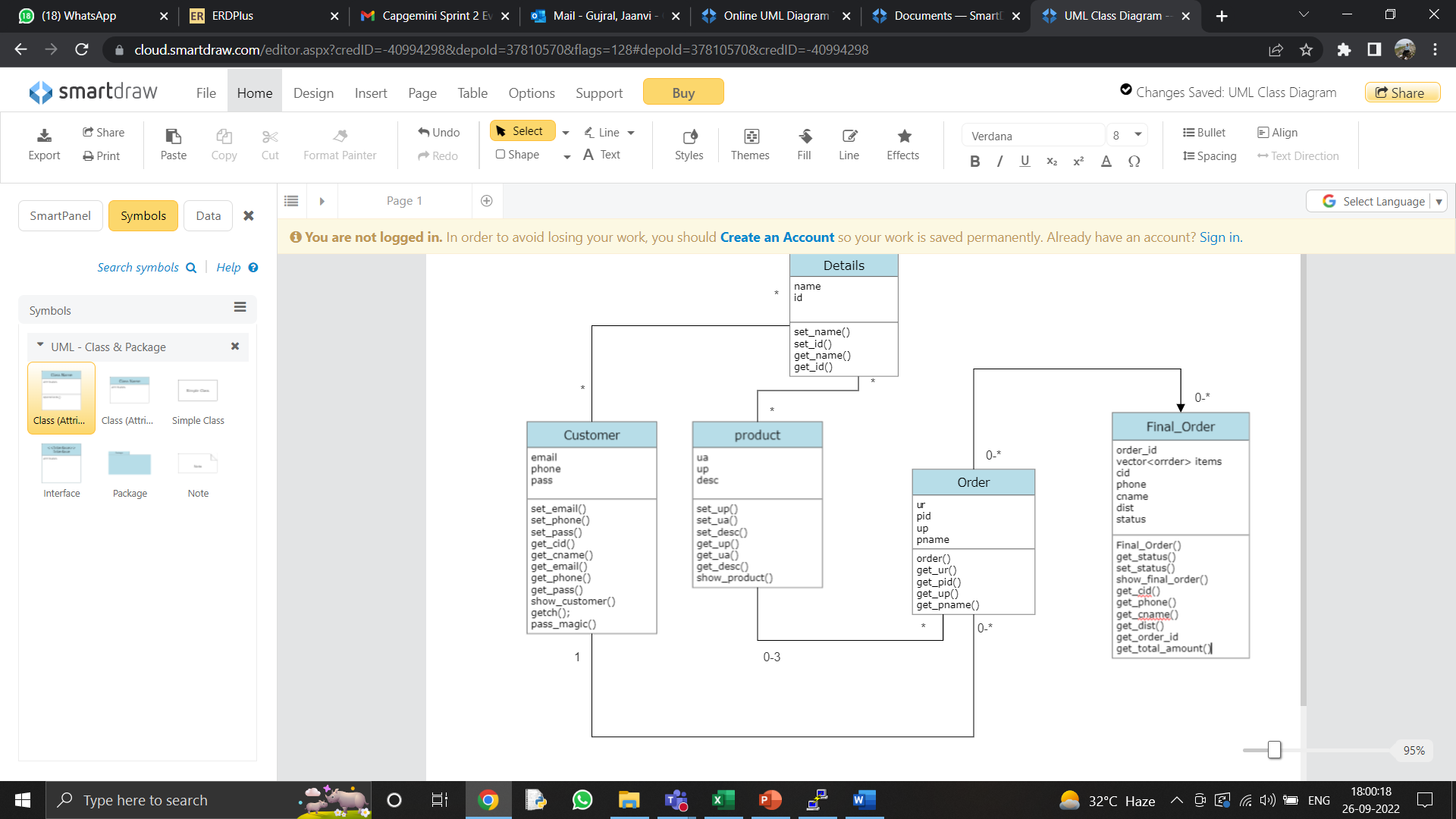
end

* **UML Diagram**

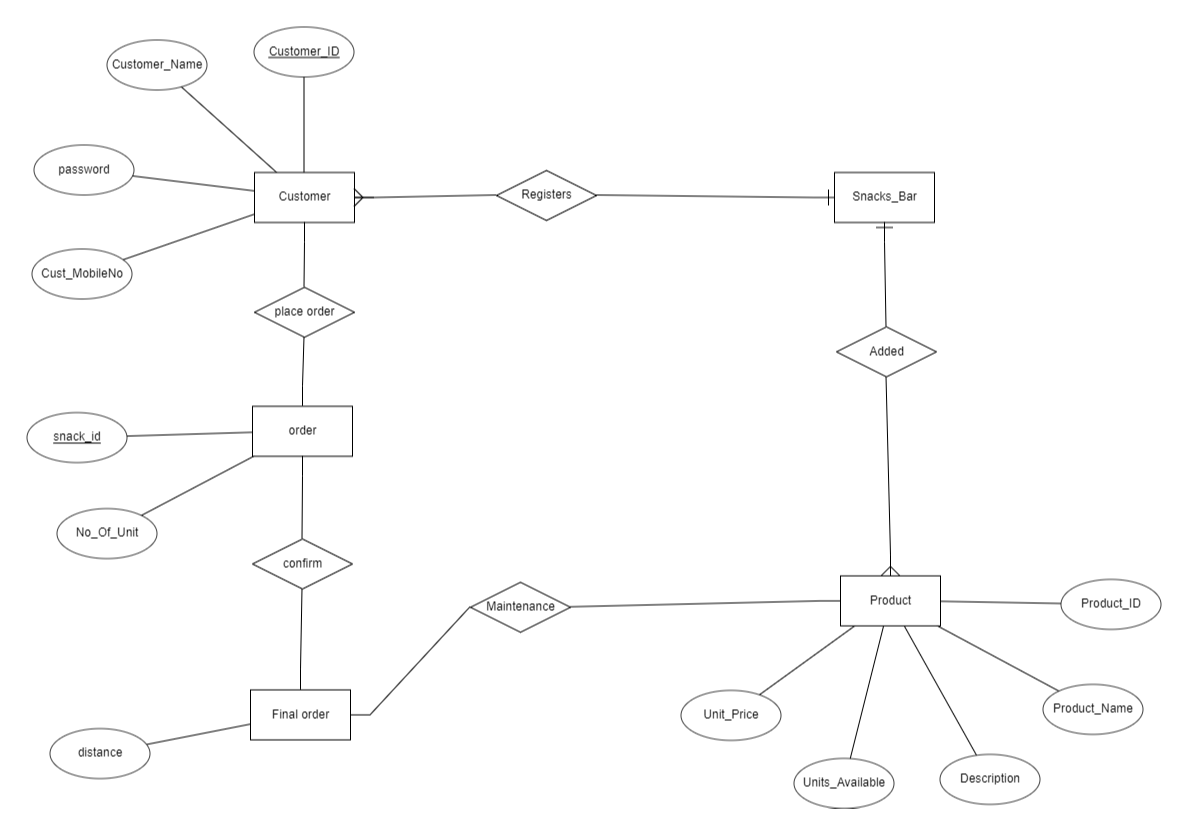
**Sequence Diagram**



**Class Diagram**

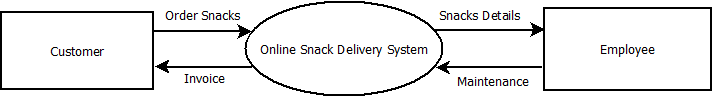


**ER Diagram**



* **Data Flow Diagram**

**Level 0**



**Level 1**

