



Topic : Online Grocery Delivery system

Group no : KDY_G8

Campus : Kandy

Submission Date : 10/14/2021

We declare that this is our own work and this Assignment does not incorporate without acknowledgment any material previously submitted by anyone else in SLIIT or any other university/Institute. And we declare that each one of us equally contributed to the completion of this Assignment.

Registration No	Name	Contact Number
IT21038778	Nissanka N.M.H.H	0705283441
IT21068850	Yusri M.A.M	0763503045
IT21124976	Rajarithne J.K.D	0777127777
IT21034268	Wimalasooriya S.A.N	0766673004
IT21041648	H.M.A.I.B Herath	0765358596

Exercise 1

- A user needs to register to the online grocery system by providing details such as name, address, password, and a security key. These details will be stored on the database with a unique ID.
- A registered customer that has forgotten their password can use forget password function to reset and add a new password.
- The store administrator will be able to add, update and delete products taken from the companies and add them to the system.
- Customers and users can leave feedback that can be a complaint or a review with the description.
- The admin can manage these feedbacks by deleting unwanted ones.
- Users and customers can also view latest news, showing stores social media pages and emails.
- The customer can view the added products and filter according to relevant categories.
- The customer then will be able to add products of their choosing to the cart that will generate a unique cart ID.
- After adding required products customers will be able view their cart showing all the added products, relevant price, quantity, and the total price.
- Customer can edit or delete products in their cart according to their choice.
- Customer can then proceed to checkout to add payment details including payment method and card details, system will validate this payment.

Rejecting

- **Redundant** – Customer, user refers to customer.
- **An event or operation** – Forgotten their password, view latest news, leave feedback, view cart, add products, edit, or delete products, proceed to checkout.
- **Outside scope of the system** - Administrator, Social media page, email, companies.
- **An attribute** – Name, address, password, security key, unique ID, new password, complaint, review, price, quantity, total price, payment method, card details, categories.

Classes

- Users
- Feedback
- Product
- Cart
- Payment

Exercise 2

User class	
Responsibilities	Collaborations
Register details	Product Feedback
View products	
Leave feedback	

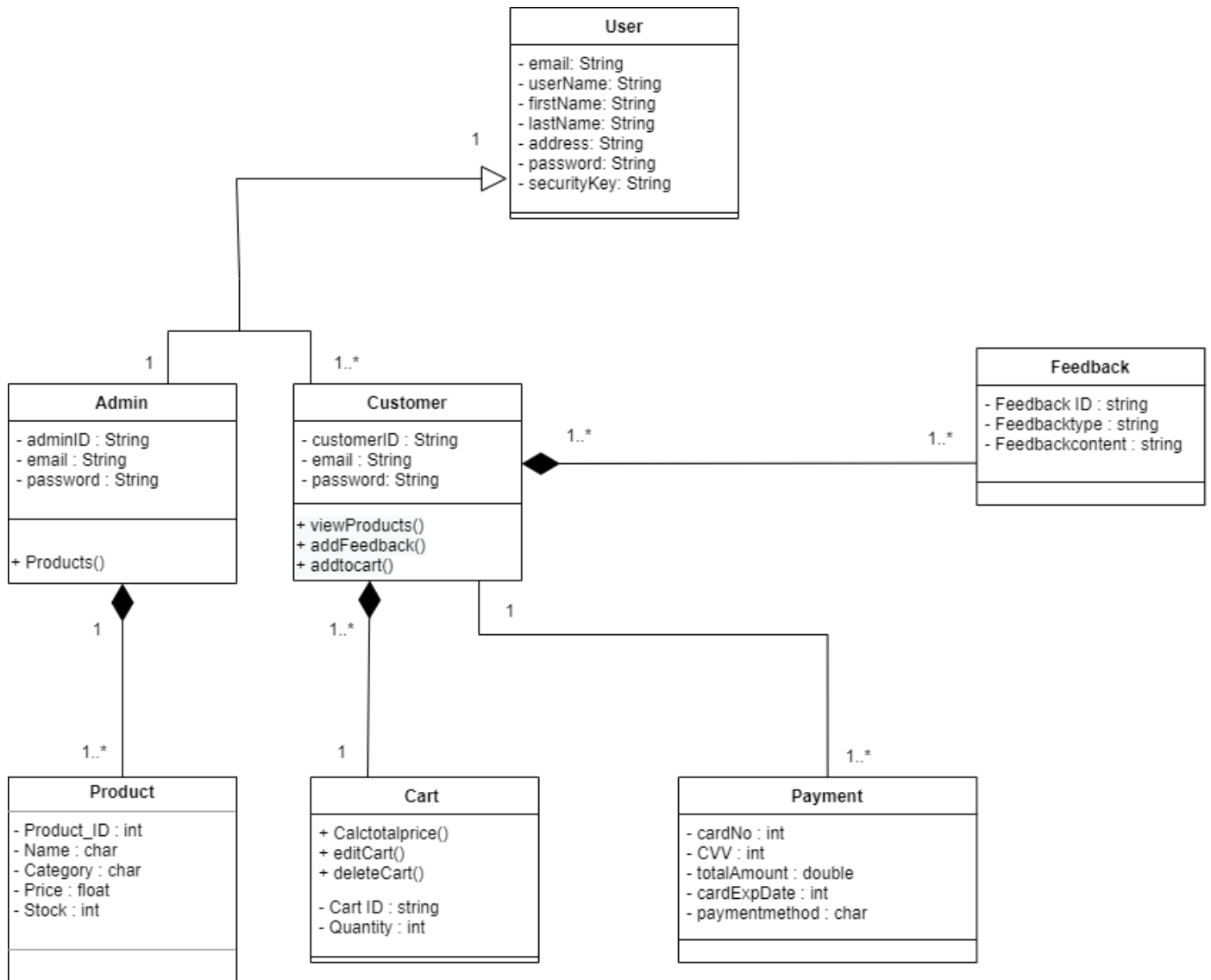
Feedback class	
Responsibilities	Collaborations
Store complaints	
Store reviews	

Product class	
Responsibilities	Collaborations
Add products	
Edit products	
Delete products	

Cart class	
Responsibilities	Collaborations
Store products	Product, User
Edit quantity	
Remove products	
Proceed to checkout	Payment

Payment class	
Responsibilities	Collaborations
Store payment details	
Validate payment	

Exercise 3



User, Admin, Customer - Rajarathne J.K.D - IT21124976

Product – Nissanka N.M.H.H – IT21038778

Cart – Yusri M.A.M - IT21068850

Payment – Wimalasooriya S.A.N - IT21034268

Feedback - H.M.A.I.B Herath - IT21041648

Exercise 4

Rajarithne J.K.D - IT21124976 Class creation

User

```
1 class User
2 {
3     protected:
4         char email[35];
5         char fName[25];
6         char lName[25];
7         char address[60];
8         char password[15];
9         char sKey[15];
10
11     public:
12         User();//construction
13         User(char uemail[], char ufName[], char ulName[], char uaddress[], char upassword[], char usKey[]);//overloaded construction for registration.
14         void setEmail(char uemail[]);//setter for email
15         int getEmail();//getter for email
16         void setFName(char ufName[]);//setter for first name
17         int getFName();//getter for first name
18         void setLName(char ulName[]);//setter for last name
19         int getLName();//getter for last name
20         void setAddress(char uaddress[]);//setter for address
21         int getAddress();//getter for address
22         void setPassword(upassword[]);//setter for password
23         int getPassword();//getter for password
24         void setKey(char usKey[]);//setter for security sKey
25         int getKey();//getter for security key
26
27
28         ~User();//destructor called for registered user
29     };
```

Admin

```
1 #include "user.h"
2 #include "product.h"
3 class admin : public User
4 {
5     protected:
6         int adminID;
7
8         Product* Products;//create product
9     public:
10         admin();//construction
11         admin(char uemail[], char ufName[], char ulName[], char uaddress[], char upassword[], char usKey[]);//overloaded construction for registration.
12         void setEmail(char uemail[]);//setter for email
13         int getEmail();//getter for email
14         void setFName(char ufName[]);//setter for first name
15         int getFName();//getter for first name
16         void setLName(char ulName[]);//setter for last name
17         int getLName();//getter for last name
18         void setAddress(char uaddress[]);//setter for address
19         int getAddress();//getter for address
20         void setPassword(upassword[]);//setter for password
21         int getPassword();//getter for password
22         void setKey(char usKey[]);//setter for security sKey
23         int getKey();//getter for security key
24
25         void Products(int prodID, char pName[], char pCategory[], double pPrice, int pStock);
26         void viewProducts();//gives admin access to view products
27         void addProducts();//gives admin access to add new products to the website
28         void updateProducts();//gives admin access to update product details and/or stock
29         void deleteProducts();//gives admin access to delete products
30
31         ~admin()
32         {
33             delete Products;//destructs products object
34         }
35     };
```

Customer

```
1  #include "Cart.h"
2  #include "Feedback.h"
3  #include "User.h"
4  #include "Product.h"
5  class customer : public User
6  {
7      protected:
8          int customerID;
9
10         Cart* Carts;//when a new user is registered, a new cart is assigned for the user.
11         Feedback* Feedback1;//when new user is registered, they are assigned with a feedback id.
12     public:
13         customer();//constructor
14         customer(char uemail[], char ufName[], char ulName[], char uaddress[], char upassword[], char usKey[]);//overloaded constructor for registration.
15         void setemail(char uemail[]);//setter for email
16         int getemail();//getter for email
17         void setfName(char ufName[]);//setter for first name
18         int getfName();//getter for first name
19         void setlName(char ulName[]);//setter for last name
20         int getlName();//getter for last name
21         void setaddress(char uaddress[]);//setter for address
22         int getaddress();//getter for address
23         void setpassword(upassword[]);//setter for password
24         int getpassword();//getter for password
25         void setsKey(char usKey[]);//setter for security sKey
26         int getsKey();//getter for security key
27
28         void viewProducts(char pName[]);//allows user to view available products
29         void viewFeedback(char Feedbacktype[], char Feedbackcontent[]);//allows user to view feedback
30         void addtoCart(char cartid[], int Qty);//allows user to add items to their cart
31         void addFeedback();//allows user to submit feedback about the website
32
33         ~customer()
34         {
35             delete Carts;//destructs cart object
36             delete Feedback1;//destructs feedback object
37         }
38     };
```

Nissanka N.M.H.H – IT21038778 Class creation

Product

```
1  class product
2  {
3      private:
4          int productID;
5          char name[100];
6          char category[100];
7          double price;
8          int stock;
9
10     public:
11         product(); //default constructor
12         product( int prodID, char pname[], char pcategory[], double pPrice, int pstock ); //overload constructor
13         void setproductdetails( int pID, char pName[], char pCategory[], double pprice, int pStock);//setter
14         int getID();//getters
15         int getName();
16         int getCategory();
17         double getPrice();
18         int getStock();
19         ~product();//destructor
20     };
```

Main

```
1  #include <iostream>
2  #include "user.h"//includes user class
3  #include "payment.h"//includes payment class
4
5  using namespace std;
6
7  int main()//start of main function
8  {
9      //create static objects
10     payment transaction_1;
11     admin admin1;
12     customer customer1;
13
14     return 0;//termination of main
15 }
```

Yusri M.A.M - IT21068850 Class creation

Cart

```
1  class Cart
2  {
3
4  private:
5      char CartID[6];
6      int quantity;
7
8  public:
9      Cart(char CID[], int qty);//Overloaded constructor
10     void setdetails(char cart[], int Qty);//Setter
11     int getqty();//Getters
12     char getid();//Getters
13     void editCart();
14     void deleteCart();
15     float Calctotalprice();
16     ~Cart();//Destructor
17 };
```


Wimalasooriya S.A.N - IT21034268 Class creation

Payment

```
1  #pragma once
2
3
4  class payment
5  {
6
7      private:
8          int CardNo;
9          int CVV;
10         float TotalAmount;
11         char CardExpDate[10];
12         char PaymentMethod[20];
13
14
15     public:
16         //default constructor
17         payment();
18
19         //overloaded constructor
20         payment(int cno ,int cvv_ ,double amount ,char cexpd[],char pm[]);
21
22         //assign card number
23         void set_cardnumber(int cno);
24         int get_cardnumber();
25
26         //assign CV number
27         void setcvv(int cvv_);
28         int getcvv();
29
30         //assign amount
31         void setamount(float amount);
32         float getamount();
33
34         //assign card expiration date
35         void setcardexpirationdate(char cexpd[]);
36         char getcardexpirationdate();
37
38         //assign payment method
39         void set_paymentmethod(char pm[]);
40         char get_paymentmethod();
41
42         //input payment details
43         void input_payment();
44
45         //generates invoice number
46         void generateinvoice();
47
48         //generate current day in (day-month-year) format
49         void generatecurrentday();
50
51         //checks the maximum value a card can exist (max = 2147483647)
52         void checkcardnumberlength();
53
54         //check card expiration date(cardexpirationdate>currentdate)
55         void checkcardexpirationdate();
56
57         //submit payment information
58         void submit();
59
60         //output necessary details
61         void output_payment();
62
63         //destructor
64         ~ payment();
65
66     };
```

H.M.A.I.B Herath - IT21041648 Class creation

Feedback

```
1  #pragmaonce
2  class Feedback
3  {
4
5  private :
6      char FeedbackID[6];
7      char feedbacktype[10];
8      char feedbackcontent[500];
9
10 public:
11     Feedback(char FeedbackID[],char feedbacktype[] ,char feedbackcontent[]);//Overloaded constructor
12     void setdetails(char FeedbackID[],char feedbacktype[] ,char feedbackcontent[]);
13     char getid();
14     char getfeedbacktype();
15     char getfeedbackcontent();
16     ~Feedback();//Destructor
17
18     };
```