Topic : Online Market Store

Group no : MLB_04.02_05

Campus : Malabe

Submission Date : 30 /05 /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**
IT20639976	Hansika A. G. D. R.	0765888479
IT20010362	Francis J. V.	0777257736
IT20622596	Weerasinghe R. B.	0712971838
IT20643454	Jayangani W. H.	0779683159

Exercise 1

1.Noun and Verb Analysis

1. All the users can search products sorting by categories.
2. Any visitor can register as a customer or a seller.
3. All the members can login to the website by providing details such as email and password.
4. Once customer and seller registered, they will be able to update their details.
5. Seller can manage the inventory system by adding products and update the products.
6. A customer can place an order by viewing the product, adding the desired product to the cart and then by making the payment.
7. The customer can select payment method according to his desire.
8. Admin can verify all the orders.
9. Inventory manager can store of orders.
10. Inventory manager can update the reports.
11. Admin generates reports.
12. Delivery person can deliver orders.
13. Delivery person updates order completion to the system.

2. Noun and Verb Analysis

Nouns- Red

Verbs - Blue

1. All the users can search products sorting by categories.
2. Any visitor can register as a customer or a seller.
3. All the members can login to the website by providing details such as email and password.
4. Once customer and seller registered, they will be able to update their details.
5. Seller receives the payment when issuing customer’s order
6. A customer can place an order by viewing the product, adding the product to the cart and then by making the payment.
7. The customer can select payment method according to his desire.
8. Admin can verify all the orders.
9. Inventory manager can store of orders.
10. Inventory manager can update the reports.
11. Admin generates reports.
12. Delivery person can deliver orders.
13. Delivery person updates order completion to the system.
14. External auditor inspects the reports every financial year.
15. Admin maintain the system.

Users	Redundant
Products	Class
Categories	Outside Scope
Visitor	Redundant
Customer	Class
Seller	Class
Members	Redundant
E mail	Attribute
Password	Attribute
Payment	Class
Order	Class
Cart	Outside Scope
Admin	Outside Scope
Inventory Manager	Outside Scope
Reports	Class
Delivery Person	Outside Scope
System	Outside Scope
External Auditor	Outside Scope
Financial year	Outside Scope

3. Noun and Verb Analysis

Classes

- Product
- Customer
- Seller
- Order
- Payment
- Reports
- Admin

Nouns

- Users
- Products
- Categories
- Visitor
- Customer
- Seller
- Members
- E mail
- Password
- Payment
- Order
- Cart
- Admin
- Inventory Manager
- Reports
- Delivery Person
- System
- External Auditor
- Financial year

Exercise 2

CRC Cards

Class Name: Product	
Responsibilities	**Collaborations**
Search Products	Customer, Seller
Viewing the product	
Adding the product	

Class Name: Customer	
Responsibilities	**Collaborations**
Register as a customer	
Login to the website	

Class Name: Seller	
Responsibilities	**Collaborations**
Register as a customer	
Login to the website	

Class Name: Admin	
Responsibilities	**Collaborations**
.Login to the system	
Maintain the system	

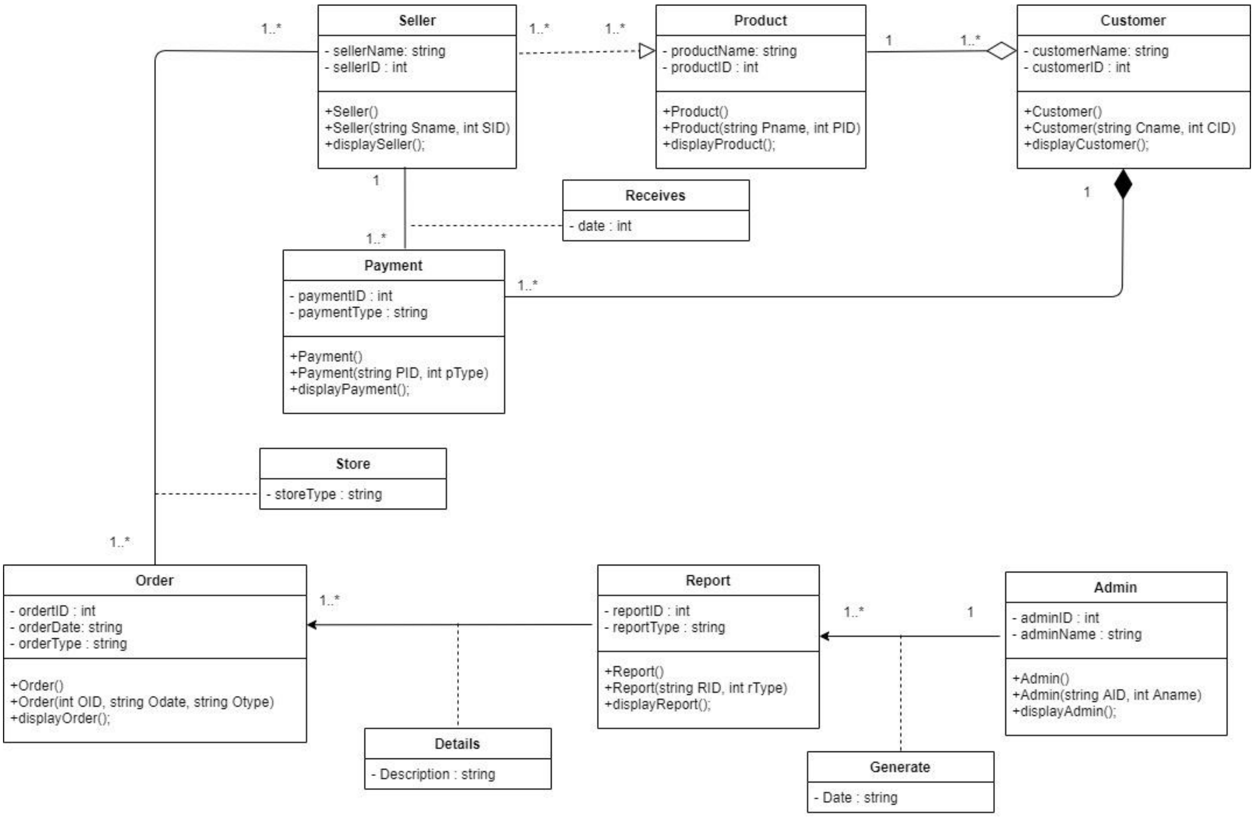
Class Name: Order	
Responsibilities	**Collaborations**
Verify Order	Admin
Store details of order	Admin, Seller
Deliver the order	
Update order	Seller

Class Name: Payment	
Responsibilities	**Collaborations**
1.Receive payments	Seller
2.Making the payment	Customer
3.Select payment	Customer

Class Name: Reports	
Responsibilities	**Collaborations**
1.Generate Report	Admin
1.Update Report	Admin
2.Inspects Report	

Exercise 3

UML Diagram



Exercise 4

Coding for the Classes

```
#include <iostream>
#include<string>
#define SIZE 5
using namespace std;




class Seller;
class Order;
class Payment;
class Customer;
class Report;
class Admin;
class Product;


//Payment Class
class Payment{
  private:
    int paymentID;
    string paymentType;
    Seller *sell[SIZE];
  public:
    Payment();
    Payment(int PID,string pType ){
      paymentID = PID;
      paymentType = pType;
    }
    void displayPayment();
    ~Payment(){
      cout << "Delete Payment" << paymentID << endl;
    }
};



//Product Class
class Product{
  private:
    string productName;
    int productID;
public:
    Product();
    Product(string pname,int PID){
      productName = pname;
      productID = PID;
    }
    void displayProduct(){
      cout << "Product Name : " << productName << endl;
      cout << "Product ID : " << productID << endl;
    }
    ~Product(){
      cout << "Delete Product" << endl;
    }
};



//Customer Class
 class Customer{
   private:
    Payment *pay[SIZE];
    string customerName;
    int customerID;
    Product *pro[SIZE];
   public:
    Customer(){
      pay[0]= new Payment(202, "Cash");
      pay[1]= new Payment(203, "Card");
    }
    Customer(string Cname,int CID){
      customerName = Cname;
        customerID=  CID;
    }
    void displayCustomer(){
      cout << "Customer Name : " << customerName << endl;
      cout << "Customer ID : " << customerID << endl;
      for (int i=0; i<SIZE; i++)
        pay[i]->displayPayment();
    for (int p=0; p<SIZE; p++)
        pro[p]->displayProduct();
    }
    ~Customer(){
      cout << "Payment Deleted" << endl;
      for (int r=0;r<SIZE;r++)
        delete pay[r];
      cout << "Everything is deleted" << endl;
    }
 };
```

```
//Order class
class Order{
  private:
    int orderID;
    string orderDate;
    string orderType;
    Seller *sell[SIZE];

  public:
    Order();
    Order(int OID, string Odate, string Otype){
      orderID = OID;
      orderDate = Odate;
      orderType =  Otype;
    }
    void displayOrder();
    ~Order(){
      cout << "Order has been deleted " << endl;
    }
};



//Seller Class
class Seller{
  private:
    string sellerName;
    int sellerID;
    Order *order[SIZE];
    Payment *pay[SIZE];

  public:
    Seller();
    Seller(string Sname, int SID){
      sellerName = Sname;
      sellerID = SID;
    }
    void displaySeller(){
      cout << "Seller Name : " << sellerName << endl;
      cout << "Seller ID : " << sellerID << endl;
      for(int k=0;k<SIZE; k++)
        order[k]->displayOrder();
      for(int k=0;k<SIZE; k++)
        pay[k]->displayPayment();
    }
    ~Seller(){
      cout <<"Delete Seller " << endl;
    }
};
```

```
//Report Class
class Report{
  private:
    int reportID;
    string reportType;
    Order *order;
  public:
    Report();
    Report (int RID, string rType){
      reportID = RID;
      reportType = rType;
    }
    void displayReport(){
      cout << "Report ID " << reportID << endl;
      cout << "Report Type " << reportType << endl;
      order->displayOrder();
    };
     ~Report(){
      cout << "Reports has been deleted " << endl;
    }
};
```

```
//Admin Class
class Admin{
  private:
    string adminName;
    int adminID;
    Report *report;
  public:
    Admin();
    Admin(string Aname, int AID){
      adminName = Aname;
      adminID = AID;
    }
    void displayAdmin(){
      cout << "Admin Name : " << adminName<< endl;
      cout << "Admin ID : " << adminID << endl;
      report->displayReport();
    }
    ~Admin(){
      cout << "Admin has been deleted " << endl;
    }
};
```

```
//Order class display function declaration
void Order:: displayOrder(){
        cout << "Ordrer ID : " << orderID << endl;
        cout << "Ordrer Date : " << orderDate << endl;
        cout << "Ordrer Type :  " << orderType << endl;
        for(int j=0; j<SIZE; j++)
          sell[j]->displaySeller();
}

//Payment class display function declaration
void Payment::displayPayment(){
      cout << "Payment ID " << paymentID << endl;
      cout << "Payment Type " << paymentType << endl;
      for(int y=0; y<SIZE; y++)
        sell[y]->displaySeller();
}

int main(){

  Customer *cus1 = new Customer ("Amal Weerasinghe",001);

  Seller *S1 = new Seller("Nimal Dias", 67546);

  Admin *A1 = new Admin("R Rajapakshe",2789);

  Order *O1 = new Order(201, "2021/05/16", "Cash");

  Product *p1 = new Product ("Detol",56876);

  Report *R1 = new Report (202, "File");

  Payment *paym = new Payment(563902561,"Card");

  cus1->displayCustomer();
  S1->displaySeller();
  A1->displayAdmin();
  O1->displayOrder();
  p1->displayProduct();
  R1->displayReport();
  paym->displayPayment();

  delete cus1;
  delete S1;
  delete A1;
  delete O1;
  delete p1;
  delete R1;
  delete paym;
  return 0;
};
```