



Topic : Online Tailor Store

Group no : KGL group 02

Campus : Kurunegala

Submission Date: 2022/05/20

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
IT21271878	Chathuranga M.K.I.S	0766753737
IT21261800	Bandara D.M.J.K.	0713154192
It21258862	Herath H.M.H.C	0712547688
It21267154	Kulathunga K.M.P.N	0714282555
It21260742	Abesingha A.H.M.Y.B	07771789166

DESCRIPTION

An online tailor store system where customers can order and get their clothes online. Clothes that customers order online, next delivered by the store by delivery agent. The delivery agent after Delivers the package to the customer. All payments here Can be made using online payment. However, all money transfer systems are analyzed and stored by the accountant. So, we can be guaranteed with our system.

Requirement

1. Any customer can go to the online tailor shop and browse the website using the URL.
2. The new user must first register with the system after registration, the registered customer must enter the username and password, after which he can access the system.
3. After the customer registers at the online tailor shop, the customer can edit the user account.
4. Customers can search for clothing by fabric type, name, and value.
5. The customer can add the fabric of his choice to the order.
6. Select the payment method after ordering the clothes. Method (Debit Card, Credit Card, Paying Friend)
7. The customer confirms the order, gives the order and the fabric is updated.
8. The financial officer validates the payment details.
9. The Finance Officer sends a confirmation message to Customer about valid payment details.
10. The system then provides customer information for distribution.
11. Then the delivery staff will check the order details and prepare to deliver.
12. User feedback that the online system administrator can add or remove users and respond to.

IDENTIFIED CLASSES

- | | |
|---------------------------|---------------------------|
| 1. Customer | - Class |
| 2. Online Tailor Store | - Outside scope of system |
| 3. Username | - Attribute |
| 4. Password | - Attribute |
| 5. Clothes | - Class |
| 6. Order list | - Redundant |
| 7. Payment | - Class |
| 8. Debit card/Credit card | - Class |
| 9. Order | - Class |
| 10.Payment details | - Attribute |
| 11.Delivery | - Class |
| 12.Users | - Redundant |
| 13.Admin | - Class |
| 14.System | - Class |
| 15.New user | - Class |
| 16.Registered Costumer | - Class |

SELECTED CLASSES

- Costumer - class
- Clothes - class
- Payment - class
- Order - class
- Delivery - class
- Admin - class
- System - Class
- New user - Class
- Registered Costumer -Class

CRC CARDS

Class Name: user	
Responsibility	Collaborators
Register to the online tailor store	system

Class Name: Clothes	
Responsibility	Collaborators
Add new clothes	Admin
Update clothes	Admin
Restore clothes	system

Class Name: Payment	
Responsibility	Collaborators
Conform payment information	Registered customer
View payment details	admin

Class Name: Order	
Responsibility	Collaborators
Accept the order	System
Cancel order	Order

Class Name: Delivery

Responsibility	Collaborators
Prepare to delivery	Order

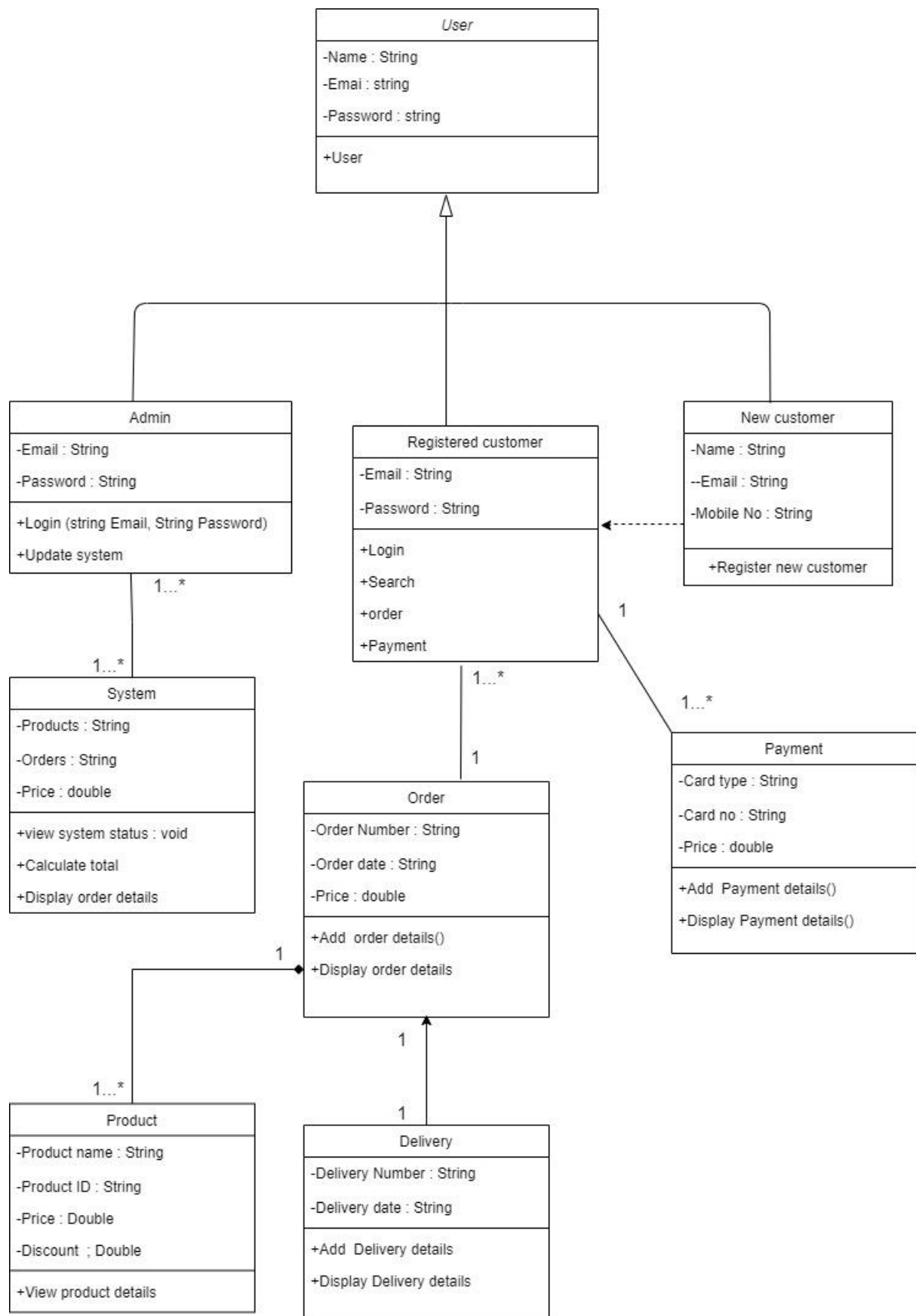
Class Name: Admin	
Responsibility	Collaborators
Operate the system	System
Manage the system	System

Class name: Registered user	
Responsibility	Collaborators
Can view their profile	System
Edit profile	Payment
Make payment	System
Select item	System

Class name: New user	
Responsibility	Collaborators
Can view store item	System
Can get item description	Product

Class name: System	
Responsibility	Collaborators
Manage the orders	Order
Manage the delivers	Delivery
Manage the products	Product

Class diagram



Exercise 2

Coding

Admin.h

```
#define SIZE[10]
#include "System.h"
#include "User.h"
#include <string>
```

```
class Admin : public User{
```

```
private:
```

```
string email;
```

```
string Password;
```

```
System* Systems[SIZE];
```

```
public:
```

```
Admin();
```

```
Admin(string pemail,string pPassword);
```

```
void login(string pemail,string pPassword);
```

```
void UpdateSystem();
```

```
~Admin();
```

```
};
```

Admin.cpp

```
#include<iostream>
```

```
using namespace std;
```

```
#include <string>
```

```
#include "Admin.h"
```

```
Admin::Admin()
```

```
{
```

```
cout << "Default Constructor Admin() called" << endl;
```

```
}
```

```
Admin::Admin(string pemail,string pPassword);
```

```
{
```

```
    email=pemail;
```

```
    Password=pPassword;
```

```
}
```

```
Admin::~~Admin()
```

```
{
```

```
    cout << "Destructed" << endl;
```

```
}
```

New customer.h

```
#include<string>

using namespace std;

class newcustomer{
protected:
    string name;
    string email;
    int mobileno;

public:

    newcustomer(string name,string email,intmobileno);
    void viewcustomerdetails();

};
```

New customer.cpp

```
#include<iostream>
#include"newcustomer.h"

newcustomer::newcustomer(){
    cout<<"default constructor newcustomer() called"<<endl;
}

newcustomer::newcustomer(string name,string email,int mobileno){
    name=name;
    email=email;
    mobileno=mobileno;

}
```

```
newcustomer::~~newcustomer(){  
    cout<<"destruacted"<<endl;  
}
```

Payment.h

```
#include <string>  
#include "RegisteredCustomer.h"  
class Payment {  
private:  
    double card number;  
    string Card type;  
    RegisteredUser* regUser;  
public:  
    Payment();  
    void PaymenDetails();  
    void displayPaymentDetails();  
    ~Payment();  
};
```

Payment.cpp

```
#include <iostream>  
#include <string>  
#include "Payment.h"  
#include "RegisteredUser.h"  
using namespace std;  
Payment::Payment() {  
    cout << "Default constructor Payment" << endl;  
}  
Payment::Payment(int id, double amt, string method,) {  
    paymentID = id;  
    amount = amt;  
    paymentMethod = method;  
}  
Payment::~~Payment() {  
    cout << "Destructor Payment" << endl;  
}
```

User.h

```

class User {
protected:
string name;
string email;
string password;
public:
User();
User(string aName, int mobileNo, string Address);
void displayCustomerDetails();
~User();
};

```

User.cpp

```

#include <iostream>
#include <string>
#include "User.h"
using namespace std;
User::User() {
cout << "Default constructor called" << endl;
}
User::User(string aName, int mobileNo, string Address)
{
name = aName;
phoneNumber = mobileNo;
email = eAddress;
}
User::~~User()
{

```

```
cout << "Destructed"<<endl;
}
```

Order.h

```
class Order{
    protected:
        char orderNumber[30];
        char orderDate[30];
        double price;

    public:
        void addOrderDetails(char number[], char date[], double price);
        void displayOrderDetails();
};
```

Order.cpp

```
#include"Order.h"
#include<cstring>
#include<iostream>
using namespace std;

void Order::addOrderDetails(char number[], char date[], double
Oprice){
    strcpy(orderNumber,number);
    strcpy(orderDate,date);
    price=Oprice;
}

void Order::displayOrderDetails(){
    cout<<"Order Number: "<<orderNumber<<endl;
    cout<<"Order Date: "<<orderDate<<endl;
```

```
    cout<<"Order Price: "<<price<<endl;
}
```

Delivery.h

```
#include <string>

using namespace std;

class Delivery {
protected:
    string delivery Number;
    string Delivery Date;

public:
    Delivery();
    Delivery(string delivery date, string delivery Number);
    void Add Delivery Details();
    void Display Delivery Details();
    ~Delivery();

    ~Delivery();
};
```

Delivery.cpp

```
#include <iostream>
```

```
#include "Delivery.h"
```

```
Delivery::Delivery() {  
    cout << "Default Delivery() called" << endl;  
}
```

```
Delivery::Delivery(string delivery date, string delivery Number )  
{  
    delivery date = delivery date;  
    delivery number = delivery number;  
}
```

```
Delivery::~~Delivery()  
{  
    cout << "Destructed" << endl;  
}
```

Registered Customer.h

```
#include <string>
```

```
using namespace std;
```

```
class RegisteredCustomer {  
    protected:  
        string Email;  
        string Password;  
  
    public:
```



```
void login(string email, string password);  
void search();  
void Order();  
void Payment();  
};
```

Registered Customer.cpp

```
#include <iostream>  
#include "Registered Customer.h"  
Registered Customer ::Registered Customer(){  
    cout << "Default Registered Customer() called" << endl;  
}  
  
Registered Customer::Registered Customer(string email, string  
password)  
{  
    Email = email;  
    Password = password;  
}  
  
{  
    cout << "Destructed" << endl;  
}
```

Product.h

```
#include <string>
```

```
using namespace std;
```

```
class Product {  
    protected:  
        string Product name;  
        string Product Id;  
        double price;;  
        double Discount;  
  
    public:  
  
        Product(int no);  
        Product(string product Id, string product Name, double price,  
double discount);  
        void viewProductDetails();  
};
```

Product.cpp

```
#include <iostream>  
#include "Product.h"
```

```
Product::Product() {  
    cout << "Default Product() called" << endl;  
}
```

```
Product::Product(string product Id, string product Name,, double  
price, double discount)  
{  
    Product Id = product Id;;
```

```
Name=name;
    price = price;
    Discount = discount;
}
```

```
Product::~~Product()
{
    cout << "Destructed" << endl;
}
```

System.h

```
#define SIZE 10
#include <string>
#include "Admin.h";

class System{

private:

    Admin* Admin[SIZE];

    string Products;
    string Orders;
    double Price;

public:

    System();
```

```
System(string pProducts, string pOrders,double pPrice);
void CalculateTotal();
void ViewSystemStatus();
void DisplayOrderDetails();
~System();
};
```

System.cpp

```
#include<iostream>
using namespace std;
#include <string>
#include"System.h"
```

```
Admin::Admin()
{
cout << "Default Constructor Admin() called" << endl;
}
```

```
Admin::Admin(string pemail,string pPassword);
{
    email=pemail;
    Password=pPassword;
}
```

```
Admin::~~Admin()
{
    cout << "Destructed" << endl;
}
```

Main.cpp

```
#include <iostream>
#include <string>
#include "User.cpp"
#include "NewCustomer.cpp"
#include "RegisteredCustomer.cpp"
#include "Admin.cpp"
#include "Product.cpp"
#include "Order.cpp"
#include "Payment.cpp"
#include "Delivery.cpp"
#include "System.cpp"

using namespace std;

int main() {

    User* user1 = new User("Amal", 0712225562, "amal@gmail.com", "amal123");
    User* user2 = new User("Kamal", 0773333333, "kamal@gmail.com", "kamal123");

    New Customer* newCustomer1 = new NewCustomer("Sirisena", 0714445546,
"sirisena@gmail.com", "sirisena123");
    NewCustomer* newCustomer2 = new NewCustomer("Somasiri", 0775574584,
"somasiri@gmail.com", "somasiri123");

    RegisteredCustomer* regCustomer1 = new
RegisteredCustomer("amarasiri@gmail.com", "amare123");
    RegisteredCustomer* regCustomer2 = new
RegisteredCustomer("dilshan@gmail.com", "dilla1234");

    Admin* admin1 = new Admin("SLM001", "Tommy@gmail.com", "tom123");
    Admin* admin2 = new Admin("SLM001", "James@gmail.com", "james123");

    Product* product1 = new Product("PR01011", "Tommy brand shirt", 4, 4975.00,
0.00);
    Product* product2 = new Product("PRo0924", "next brand trouser", 2, 7850.00,
5.00);

    Order* order1 = new Order("ORD045", "pvt order", "2022-05-019", 4, 4975.00,
0.00);
    Order* order2 = new Order("ORD047", "Company order", "2022-05-18", 10,
31500.00, 5.00);
```

```
Payment* payment1 = new Payment("PAY041", "ORD045", "Cash on delivery",
4975.00);
Payment* payment2 = new Payment("PAY078", "ORD047", "Online", 31500.00);

Delivery* delivery1 = new Delivery("DEL045", "ORD045", "DRV002",
"Kurunegala", "Cash on delivery", 4975.00);
Delivery* delivery2 = new Delivery("DEL047", "ORD047", "DRV005",
"wariyapola", "Online", 31500.00);

System* system = new System();

delete user1;
delete user2;

delete newCustomer1;
delete newCustomer2;

delete regCustomer1;
delete regCustomer2;

delete admin1;
delete admin2;

delete product1;
delete product2;

delete order1;
delete order2;

delete payment1;
delete payment2;

delete delivery1;
delete delivery2;

delete system;

std::cout << "Main file finished.\n";
return 0;
}
```

Registration No	Name	Class Diagram	Codings
IT21271878	Chathuramga M.K.I.S	<ul style="list-style-type: none"> • New customer • product 	<ul style="list-style-type: none"> • New customer.h • New customer.cpp • Product.h • Product.cpp
IT21261800	Bandara D.M.J.K	<ul style="list-style-type: none"> • User • Payment 	<ul style="list-style-type: none"> • User.h • User.cpp • Payment.h • Payment.cpp
IT21258862	Herath H.M.H.C	<ul style="list-style-type: none"> • Admin • System 	<ul style="list-style-type: none"> • Admin.h • Admin.cpp • System.h • System.cpp
IT21267154	Kulathunga K.M.P.N	<ul style="list-style-type: none"> • order • Delivery 	<ul style="list-style-type: none"> • Order.h • Order.cpp • Delivery.h • Delivery.cpp
IT21260742	Abesinghe A.H.M.Y.B		

