



Topic : Online Salon Booking System

Group no : IWT_KDY_06

Campus : Kandy

Submission Date:

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
IT21261428	Gunawardana D.R.I.P.	077-0851557
IT21261282	Sandaru T.H.K.	072-2119798
IT21213380	Godamunna D.R.B.N.B.	075-2701838
IT21268526	Jayarathna J.A.I.S.	074-0443439
IT21215506	Basnayaka B.M.M.S.	070-3735275

System Requirements:

- User can register to the system by filling up the registration form with necessary details.
- Registered User can log into the system, using his/her username and password.
- Both Registered and Unregistered Users can view styles/products, Search products and View Reviews.
- Unregistered User must create an account to buy products from the Online Store or to make a reservation.
- Registered User can make reservations.
- Registered User can add/remove items to the cart and Pay and Confirm the order.
- Registered User can specify the payment method. E.g.: Credit Card, Debit Card, PayPal, and Cash on Delivery.
- Once the Order is Confirmed by the Registered User, Payment is validated, the order is placed, and items are updated.
- Registered User can write Reviews and Feedbacks.
- Admin must be able to add/remove items and styles to/from the system.
- Admin must be able to check reservations and assign artists accordingly.
- Admin can check reviews, inquiries and reply to them accordingly.
- Admin must be able to generate reports.

Noun:

- | | |
|-----------------------|--------------|
| 1. User - | Class |
| 2. System- | Out of scope |
| 3. Registered User- | Class |
| 4. Username- | Attribute |
| 5. Password- | Attribute |
| 6. Styles- | Class |
| 7. Products- | Class |
| 8. Reviews- | Class |
| 9. Unregistered User- | Redundant |
| 10. User Account- | Class |
| 11. Reservation- | Class |
| 12. Items- | Redundant |
| 13. Cart- | Class |
| 14. Payment Method- | Attribute |
| 15. Payment- | Class |
| 16. Order- | Class |
| 17. Admin- | Class |
| 18. Report- | Class |

Identified Classes Using Noun Verb Analysis:

1. User
2. Registered User
3. Styles
4. Products
5. Reviews
6. User Account
7. Reservation
8. Cart
9. Payment
10. Order
11. Report
12. Admin

CRC Crads:

Class name: User	
Responsibilities	Collaborations
Register to the system	User Account
Search products	Products
View Reviews	Reviews

Class name: Registered User	
Responsibilities	Collaborations
Log-In to system	User Account
Make Reservations	Reservation
Buy products	Report
Write Reviews	Review
Place an order	Order
Make a payment	Payment

Class name: Styles	
Responsibilities	Collaborations
Add styles	Admin
Remove Styles	Admin
Update details	Admin

Class name: Products	
Responsibilities	Collaborations
Add Items	Admin
Remove Items	Admin
Update Item details	Admin
Payment Details	Payment

Class name: Reviews	
Responsibilities	Collaborations
Receive feedbacks/ratings	Registered User
Check inquiries	Admin
Display feedbacks/ratings	

Class name: User Account	
Responsibilities	Collaborations
Get registration details	
Provide username & password	

Class name: Cart	
Responsibilities	Collaborations
Add/ remove items	Products
Number of items	

Class name: Payment	
Responsibilities	Collaborations
Select payment method	Registered Customer
View Payment Details	
Validate payment	

Class name: Order	
Responsibilities	Collaborations
Place Order	Registered Customer
View Order Details	Registered Customer
Validate payment	Payment

Class name: Report

Responsibilities	Collaborations
View Payment History	Admin
View placed order	Admin/Registered Customer
View remaining products	Admin

Class name: Reservation

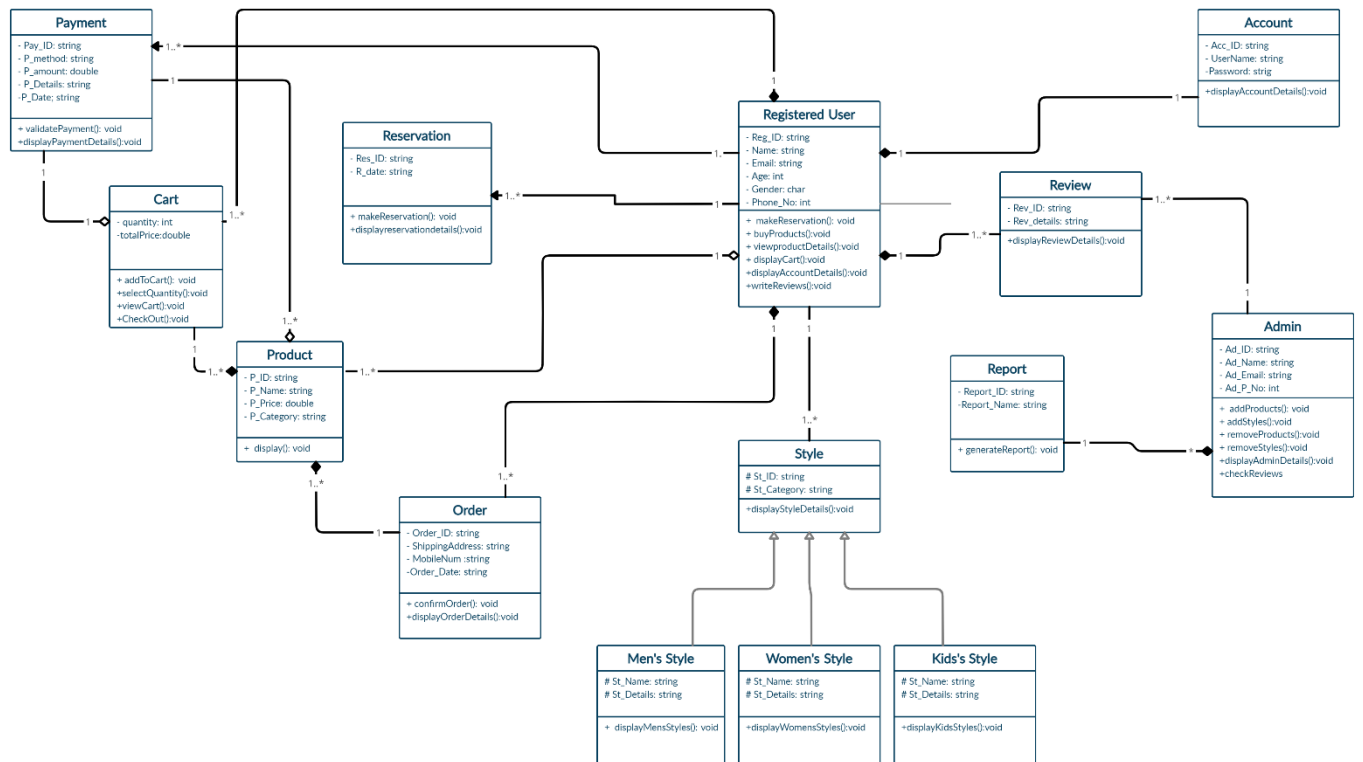
Responsibilities	Collaborations
Make Reservation	Registered Customer
Check Reservations	Admin

Class name: Admin

Responsibilities	Collaborations
Add/Remove/Update Styles	Styles
Add/Remove/Update Products	Products
Check Reviews	Reviews
Generate Reports	Reports
Check Reservations	Reservation

Class Diagram:

Online Salon Boking System



Classes and Constructors :

```
class Registered_User
{
    private:
        char Reg_ID[10];
        char Name[40];
        char Email[40];
        int Age;
        char Gender[10];
        int phoneNo;

    public:
        Registered_User();
        Registered_User(const char Reg_ID[], const char Uname[], const char
Uemail[], int Uphone[], const char Gender[], int age);
        void displayDetails();
        ~Registered_User();
};
```

```
class Account
{
    private:
        char Acc_ID[10];
        char UserName[40];
        char Password[40];

    public:
        Account();
        Account(const char Acc_ID[], const char UserName[], const char
Password[]);
        void displayDetails();
        ~Account();
};
```

```
class Style
{
    private:
        char Style_ID[10];
        char Style_catengory[100];

    public:
        style();
        style(const char style_ID[], const char Style_category[]);
        void displayCatengory();
        ~style();
};
```



```

class Reservation
{
    private:
        char Res_ID[10];
        char R_Date[20];

    public:
        Reservation();
        Reservation(const char Res_ID[], const char R_date[]);
        void displayDetails();
        ~Reservation();
};

class Admin
{
    private :
        string Ad_ID;
        string Ad_Name;
        string Ad_Email;
        int Ad_P_no;
    public :
        Admin();
        Admin(int aID, string aName, string aEmail, Report *RPT);
        void Admin::displayAdminDetails()
        ~Admin();
};

```

```

class Order
{
    private:
        char Order_ID[10];
        char Shipping Address[50];
        char MobileNum[10];
        char Order_Date[20]
    public :
        Order();
        Order(string Order_ID, string Shipping Address, string MobileNum,
string Order_Date);
        void displayOrderDetails();
        ~Order();
};

```

```

class Cart
{
    private:
        int quantity;
        double totalPrice;

    public :
        Cart();
        Cart(int quantity, double totalPrice);
        void displayDetails();
        ~Cart();
};

```

```

class Product
{
    Private:
        string P_ID;
        string Name;
        double P_Price;
        string P_Category;

    Public:
        Product();
        Product( string P_ID,string Name,double P_Price,string P_Category);
        void displayProductetails();
        ~Product();
};

```

```

class Payment
{
    Private:
        string Pay_ID;
        string P_method;
        double P_amount;
        string P_Details;
        string P_date;

    Public:
        Payment();
        Payment(string Pay_ID,string P_method,double P_amount,string
P_Details,string P_date);
        void displayPaymentetails();
        ~Payment();
};

```

```
class Report
{
    private:
        char Rep_ID[10];
        char Rep_Details[100];

    public:
        Report();
        Report(const char Rep_ID[], const char R_Details[]);
        void displayDetails();
        ~Report();
};
```

```
class Review
{
    private:
        char Rev_ID[10];
        char Rev_Details[50];

    public:
        Review();
        Review(const char Rev_ID[], const char Rev_Details[]);
        void displayDetails();
        ~Review();
};
```

Main Program:

```
#include <iomanip>
#include <cstring>

#include <Registered_User.h> //Header files
#include <Account.h>
#include <Style.h>
#include <Reservation.h>
#include <Admin.h>
#include <Order.h>
#include <Cart.h>
#include <Product.h>
#include <Payment.h>
#include <Report.h>
#include <Review.h>

using namespace std;

int main () //Main Program
{
    Registered_User * reg1;
    reg1 = new Registered_User();

    Account * Acc1;
    Acc1 = new Account();

    Style * st1;
    st1 = new Style();

    Reservation * res1;
    res1 = new Reservation();

    Admin * ad1;
    ad1 = new Admin();

    Order * od1;
    od1 = new Order();

    Cart * ct1;
    ct1 = new Cart();

    Product * pdc1;
    pdc1 = new Product();

    Payment * pay1;
    pay1 = new Payment();
```

```
Report * rp1;  
rp1 = new Report();  
  
Review * rev1;  
rev1 = new Review();  
  
return 0;  
}
```