



Topic : Online Auction System

Group no : MLB_06.01_07

Campus : Malabe

Submission Date : 20 May 2022

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
IT21318320	Silva T.U.D	076 900 5515
IT21321054	Bandara D.M.E.K	076 944 7415
IT21319242	Ilangerathne R.A.C.W	071 543 1391
IT21318948	Dissanayake K.B	076 072 4229
IT21313684	Karunarathna D.T.S	071 161 1308

CRC Cards

- Message – IT21318320 – Silva T.U.D

Class name: Message	
Responsibilities	Collaborations
Send Message	Registered User
Save Message	
Display Message	

- Issue – IT21318320 – Silva T.U.D

Class name: Issue	
Responsibilities	Collaborations
Send Issue	Message
Save Issue Type	
Save Issue	Message
Display Issue	

- Feedback – IT21318320 – Silva T.U.D

Class name: Feedback	
Responsibilities	Collaborations
Send Feedback	Message
Save Feedback	Message
Save Rating	
Display Rating	

- Payment – IT21318320 – Silva T.U.D

Class name: Payment	
Responsibilities	Collaborations
Save Details	Item
Display Details	
Validate Details	
Authorize Payment	
Display Invoice	Invoice

- Invoice – IT21318320 – Silva T.U.D

Class name: Invoice	
Responsibilities	Collaborations
Save Invoice Details	Payment
Display Details	
Print Invoice	

- Message – IT21318320 – Silva T.U.D

//Class definition of the message class.

```
class Message
```

```
{
```

```
protected:
```

```
int messageID;
```

```
char messageType[50];
```

```
Registered_user* Ruser;
```

```

public:
    Message();
    Message(int pmessageID, char pmessageType[], Registered_user *pRuser);
    void saveMessage();
    void diplayMessage();
    ~Message();
};

// Implementation of the methods of the class message.
Message::Message(){
    messageID = 0;
    strcpy(messageType, "");
}

Message::Message(int pmessageID, char pmessageType[], Registered_user *pRuser){
    messageID = pmessageID;
    strcpy(messageType, pmessageType);
    Ruser = pRuser;
}

void Message::saveMessage() {}

void Message::diplayMessage() {}

Message::~Message() {}

```

- **Issue – IT21318320 – Silva T.U.D**

//Class definition of the issue class.

```
class Issue :public Message
{
    private:
        char status[10];

    public:
        Issue();
        Issue(int cMessageID, char cMessageType[], char pstatus[]);
        void saveIssueType();
        void displayStatus();
        ~Issue();
};
```

// Implementation of the methods of the class issue.

```
Issue::Issue() {
    messageID = 0;
    strcpy(messageType, "");
    strcpy(status, "");
}

Issue::Issue(int cMessageID, char cMessageType[], char pstatus[]) {
    messageID = cMessageID;
    strcpy(messageType, cMessageType);
    strcpy(status, pstatus);
}
```

```
void Issue::saveIssueType() {}
```

```
void Issue::displayStatus() {}
```

```
Issue::~~Issue() {}
```

- **Feedback – IT21318320 – Silva T.U.D**

```
//Class definition of the feedback class.
```

```
class Feedback :public Message
```

```
{
```

```
private:
```

```
    char rating[20];
```

```
public:
```

```
    Feedback();
```

```
    Feedback(int cMessageID, char cMessageType[], char prating[]);
```

```
    void saveRating();
```

```
    void displayRating();
```

```
    ~Feedback();
```

```
};
```

```
// Implementation of the methods of the class feedback.
```

```
Feedback::Feedback() {
```

```
    messageID = 0;
```

```
    strcpy(messageType, "");
```

```
    strcpy(rating, "");
```

```
}
```

```
Feedback::Feedback(int cMessageID, char cMessageType[], char prating[]) {  
    messageID = cMessageID;  
    strcpy(messageType, cMessageType);  
    strcpy(rating, prating);  
}
```

```
void Feedback::saveRating() {}
```

```
void Feedback::displayRating() {}
```

```
Feedback::~~Feedback() {}
```

- **Payment – IT21318320 – Silva T.U.D**

//Class definition of the payment class.

```
class Payment {  
    private:  
        int paymentId;  
        float payamount;  
        char type[20];  
        int cardnumber;  
        int expdate;  
        char cardholdername[50];  
        Item* item;
```

```

    Invoice* invoiceNo[100];

public:
    Payment();
    Payment(int no1);
    void setpaymentDetails(int ppaymentId, float ppayamount, char ptype[], int pcardnumber,
int pexdate, char pcardholdername[]);
    void displayDetails();
    void validateDetails();
    void authorizePayment();
    void displayInvoice();
    ~Payment();
};

```

// Implementation of the methods of the class payment.

```

Payment::Payment() {
    invoiceNo[0] = new Invoice(001);
    invoiceNo[1] = new Invoice(002);
}

```

```

Payment::Payment(int no1){
    invoiceNo[2] = new Invoice(no1);
}

```

```

void Payment::setpaymentDetails(int ppaymentId, float ppayamount, char ptype[], int
pcardnumber, int pexdate, char pcardholdername[]) {
    paymentId = ppaymentId;
    payamount = ppayamount;
    strcpy(type, ptype);
    cardnumber = pcardnumber;
}

```



```
    expdate = pexdate;
    strcpy(cardholdername, pcardholdername);
}
```

```
void Payment::displayDetails() {}
```

```
void Payment::validateDetails() {}
```

```
void Payment::authorizePayment(){}
```

```
void Payment::displayInvoice(){
    for(int count=0; count < 100; count++)
    {
        invoiceNo[count] -> display();
    }
}
```

```
Payment::~~Payment(){
    for(int count=0; count < 100; count++)
    {
        delete invoiceNo[count];
    }
}
```

- **Invoice – IT21318320 – Silva T.U.D**

//Class definition of the invoice class.

```
class Invoice {  
    private:  
        int invoiceNumber;  
        float paidamount;  
        int itemID;  
        int userId;  
  
    public:  
        Invoice();  
        Invoice(int no);  
        void setInvoiceDetails(float ppaidamount, int pitemID, int puserId);  
        void printInvoice();  
        void display();  
        ~Invoice();  
};
```

// Implementation of the methods of the class invoice.

```
Invoice::Invoice(){  
    invoiceNumber = 0;  
    paidamount = 0;  
    itemID = 0;  
    userId = 0;  
}
```

```
Invoice::Invoice(int no) {  
    invoiceNumber = no;
```

```
}
```

```
void Invoice::setInvoiceDetails(float ppaidamount, int pitemID, int puserId){  
    paidamount = ppaidamount;  
    itemID = pitemID;  
    userId = puserId;  
}
```

```
void Invoice::printInvoice() {}
```

```
void Invoice::display() {}
```

```
Invoice::~~Invoice() {}
```

- **Main Program**

```
#include <iostream>  
using namespace std;  
  
int main(){  
  
    // Create object of class Message.  
    Message msg1;  
  
    // Create object of class Issue.  
    Issue issue1;  
  
    // Create object of class Feedback.  
    Feedback feedback1;
```

```
// Create object of class Invoice inside the the class Payment.  
Payment *payment1;  
  
payment1 = new Payment(1);  
payment1 -> displayInvoice();  
  
return 0;  
}
```