

BSc (Hons) in Information Technology**Object Oriented Concepts – IT1050****Assignment 2****Year 1, Semester 2****2023-January**

Topic : Online Customer Support System

Group no : MLB_14.01_05

Campus : Malabe

Submission Date : 13/06/2023

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
IT22320582	Jayasundara D.W.S	0764479005
IT22305350	Rajapakshe P.H.Y.L	0726998321
IT22341136	M.A.S Gunathilaka	0768777252
IT22315632	R.N.D K.Rajapaksha	0711278483
IT22337412	D.Rajapaksha	0761785646

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

Requirements for the online customer support system

1. Unregistered user can register to the system by creating an account. To create an account the visitor should enter their full name, email address, date of birth.
2. After registration, registered user have to log in to the account.
3. To access customer support resources registered user have to buy the required service plan. registered user can choose basic, premium or ultimate as service plan type.
4. In order to purchase the service plan registered user, have to select their preferred payment method as PayPal or credit card or debit card.
5. The registered user can access the knowledge base, customer support services.
6. The registered user can view and edit their profile.
7. To get the customer support services registered user have to raise a ticket. Registered user has to mention their support inquiry type, name in the ticket.
8. Upon rising ticket, registered user will be given a ticket ID number.
9. Admin of the system will assign the tickets to appropriate customer support agents.
10. Customer support agents are assigned to specific departments.
11. The registered user can track their support ticket.
12. The registered user can post feedback to improve system services.
13. The administrator can add/remove users, assign support tickets, review feedbacks, assign support tickets and generate reports.
14. The manager of the system can review reports and supervise customer support agents.
15. The developer of the system should be able to add new features, debug the system and maintain the security level of the system.

BSc (Hons) in Information Technology
Object Oriented Concepts – IT1050
Assignment 2

Classes and Noun Verb analysis

1. **Unregistered User:** Register
2. **Registered User:** login to account
3. **Account:** View Account, Edit Account
4. **Support Ticket:** Raise a Ticket, Track Ticket
5. **Customer Support Recourses:** Accessing knowledgebase, Accessing Support Services
6. **Service Plan:** purchase plan
7. **Payment:** Manage Payment (Get/Store), Validate
8. **Feedback:** Manage feedback (Submit/Get/Review)
9. **Customer Support Agent:**

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

CRC cards

User	
Responsibilities	Collaborations
Store details user	
View the pages	
Logging to the Account	Account

Customer	
Responsibilities	Collaborations
Purchase a plan	Service plan
Access support resources	Customer support resources
Provide feedback details	Feedback

Account Class	
Responsibilities	Collaborations
Store details of account	
View Account details	Registered User
Edit Account	

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

Support Ticket Class	
Responsibilities	Collaborations
Raise a Ticket	Registered User
Track the Ticket	
Responding to ticket	Support agent

Customer Support Recourses	
Responsibilities	Collaborations
Manage resources	Support agent

Service Plan	
Responsibilities	Collaborations
Store Service plan details	Registered user
Manage different service plan types	Registered user

Payment	
Responsibilities	Collaborations
Make the Payment	Registered User
Validate the payment	

BSc (Hons) in Information Technology
Object Oriented Concepts – IT1050
Assignment 2

Feedback	
Responsibilities	Collaborations
Store feedback details	
Display feedback details	Registered User

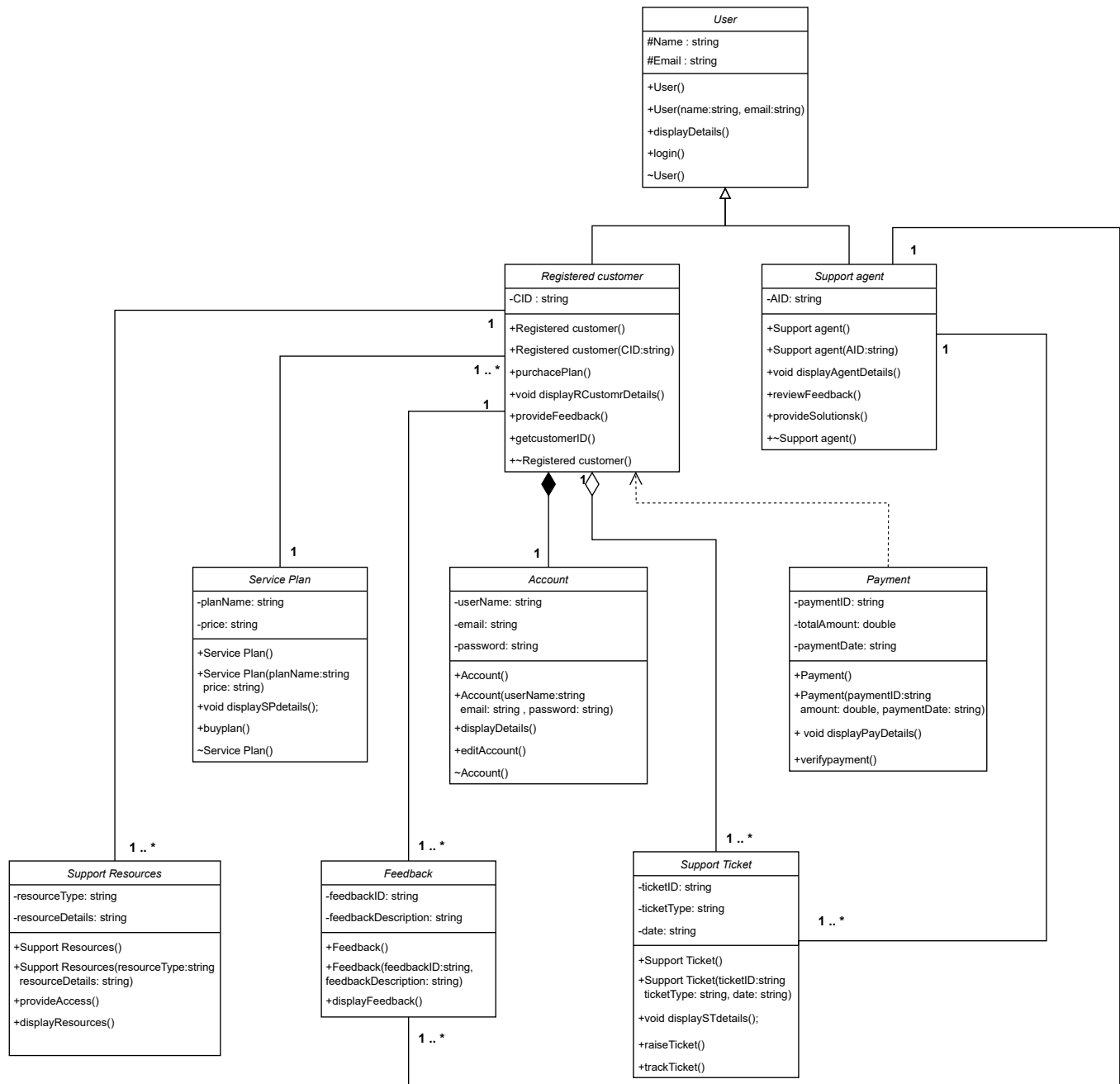
Support agent	
Responsibilities	Collaborations
Provide solutions	Ticket
Review feedback	Feedback

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

Class diagram



BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

Classes

```
#include <iostream>
#include <string>
#define SIZE 2
using namespace std;

// Forward declarations
class SupportResources;
class Feedback;
class ServicePlan;
class SupportAgent;
class RegisteredCustomer;

//User Class

class User
{
protected:
    string name;
    string email;

public:
    User();
    User(string* uName, string* uEmail);
    void displayDetails();
    void login();
};

//Account Class

class Account
{
private:
    string userName;
    string email;
    string password;

public:
    Account();
    Account(string uName, string uEmail, string uPw);
    void displayAccDetails();
    void editAccount();
    ~Account();
};
```


BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

```
//Support ticket class
class SupportTicket {
private:
    string ticketID;
    string ticketType;
    string date;
    SupportAgent* sAgent;

public:
    SupportTicket();
    SupportTicket(string tid, string ttype, string sdate , SupportAgent *csAgent);
    void displaySTdetails();
    void raiseTicket();
    void trackTicket();
    ~SupportTicket();
};
```

```
// ServicePlan Class
class ServicePlan {
private:
    string planName;
    float price;
    RegisteredCustomer* rcustomer[SIZE];

public:
    ServicePlan();
    ServicePlan(string pn, float pprice);
    void addregisteredCustomer(RegisteredCustomer* cust);
    void displaySPdetails();
    void buyPlan();
    ~ServicePlan();
};
```

```
ServicePlan::ServicePlan()
{
    planName = "";
    price = 0;
}
```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

//RegisteredCustomer Class

```
class RegisteredCustomer : public User
{
private:
    string customerId;
    Account* acc[SIZE];
    SupportResources* resources[SIZE];
    ServicePlan* sPlan;
    Feedback* fback[SIZE];
    ServicePlan* servicePlan;
    SupportTicket* sTicket[SIZE];
    int nooffd;

public:
    RegisteredCustomer();
    RegisteredCustomer(string* uName, string* uEmail, string* cid,
                       string uName1, string uEmail1, string uPw1 , ServicePlan* sPalan);
    void addSupportTicket(SupportTicket *sTicket1);
    void addResources(SupportResources* r);
    void addFeedback(Feedback* fb);
    void displayRCustomrDetails();
    void purchasePlan();
    string getCustomerId();
    void displayAccount();
    void provideFeedback();
    ~RegisteredCustomer();
};
```

//Support Agent Class

```
class SupportAgent : public User
{
private :
    string agentId;
    Feedback* fback[SIZE];
    SupportTicket* sTicket[SIZE];
    int nooffd;

public:
    SupportAgent();
    SupportAgent(string* uName, string* uEmail , string* aId);
    void addSupportTicket(SupportTicket *sticket);
    void displayAgentDetails();
    void seeFeedback(Feedback* fb);
    void reviewFeedback();
    string provideSolution();
    ~SupportAgent();
};
```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

```
// SupportResources Class
class SupportResources
{
private:
    string resourcesType;
    string resourcesDetails;
    RegisteredCustomer* cus;

public:
    SupportResources();
    SupportResources(const string& KBresourcesType,
                    const string& KBresourcesDetails, RegisteredCustomer* pcus);
    void provideAccess();
    void displayResources();
};
```

```
// Feedback Class
class Feedback
{
private:
    string feedbackId;
    string feedbackDescription;
    RegisteredCustomer* customer;
    SupportAgent* agent;

public:
    Feedback();
    Feedback(const string& fId, const string& fDescription,
            RegisteredCustomer* pcus, SupportAgent* pagent);
    void displayFeedback();
};
```

```
//Payment Class

class Payment
{
private :
    string paymentID;
    double totalAmount;
    string paymentDate;
    RegisteredCustomer *p1;

public:
    Payment();
    Payment(string payID, double amut, string payDate);
    void displayPayDetails(RegisteredCustomer* p1);
    void verifyPayment();
};
```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

Implementation of classes

//User Class Methods Implementation

```
User::User()
{
    name = "";
    email = "";
}

User::User(string* uName, string* uEmail)
{
    name = *uName;
    email = *uEmail;
}

void User::displayDetails()
{
    cout << "Name: " << name << endl;
    cout << "Email: " << email << endl;
}
}
```

//Account Class Methods Implementation

```
Account::Account()
{
    userName = "";
    email = "";
    password = "";
}

Account::Account(string uName, string uEmail, string uPw)
{
    userName = uName;
    email = uEmail;
    password = uPw;
}

void Account::displayAccDetails()
{
    cout << "Account User Name :" << userName << endl;
    cout << "Account Email :" << email << endl;
    cout << "Account Password :" << password << endl;
    cout << endl << "*****" << endl;
}

Account::~~Account()
{
    cout << "Account Deleted..." << endl;
}
}
```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

```
//RegisteredCustomer Class Methods Implementation
RegisteredCustomer::RegisteredCustomer()
{
    customerId = "";
}

RegisteredCustomer::RegisteredCustomer(string* uName, string* uEmail,
                                       string* cid, string uName1,
                                       string uEmail1, string uPw1,
                                       ServicePlan* sPlan) : User(uName, uEmail)
{
    customerId = *cid;
    acc[0] = new Account(uName1, uEmail1, uPw1);
    sPlan->addregisteredCustomer(this);
}

void RegisteredCustomer::displayRCustomerDetails()
{
    cout << "Customer ID : " << customerId << endl;
    cout << endl << "*****" << endl;
}

//Support Agent Class Methods Implementation

SupportAgent::SupportAgent()
{
    agentId = "";
}

SupportAgent::SupportAgent(string* uName, string* uEmail, string* aId) : User(uName, uEmail)
{
    agentId = *aId;
}

void SupportAgent::seeFeedback(Feedback* fb) {
    if (nooffd < SIZE) {
        fback[nooffd] = fb;
        nooffd++;
    }
}

void SupportAgent::displayAgentDetails()
{
    cout << "Agent ID : " << agentId << endl;
    cout << endl << "*****" << endl;
}

SupportAgent::~SupportAgent()
{
    cout << "Agent Details Deleted..." << endl;
    cout << endl << "*****" << endl;
}

void SupportAgent::addSupportTicket(SupportTicket *sticket)
{

```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

//Payment Class Methods Implementation

```
Payment::Payment()
{
    paymentID = "";
    totalAmount = 0;
    paymentDate = "";
}

Payment::Payment(string payID, double amut, string payDate)
{
    paymentID = payID;
    totalAmount = amut;
    paymentDate = payDate;
}

void Payment::displayPayDetails(RegisteredCustomer* p1)
{
    cout << "Payment ID :" << paymentID << endl;
    cout << "Total Amount :" << totalAmount << endl;
    cout << "Payment Date :" << paymentDate << endl;
    cout << "Customer ID :" << p1->getCustomerId() << endl;
    cout << endl << "*****" << endl;
}
```

//Feedback Class Methods Implementation

```
Feedback::Feedback()
{
    feedbackId = "";
    feedbackDescription = "";
    customer = nullptr;
    agent = nullptr;
}

Feedback::Feedback(const string& fId, const string& fDescription,
    RegisteredCustomer* pcus, SupportAgent* pagent)
{
    feedbackId = fId;
    feedbackDescription = fDescription;
    customer = pcus;
    agent = pagent;
    customer->addFeedback(this);
    agent->seeFeedback(this);
}
```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

//SupportResources Class Methods Implementation

```
SupportResources::SupportResources()
```

```
{  
    resourcesType = "";  
    resourcesDetails = "";  
    cus = nullptr;  
}
```

```
SupportResources::SupportResources(const string& KBresourcesType,  
                                   const string& KBresourcesDetails, RegisteredCustomer* pcus)
```

```
{  
    resourcesType = KBresourcesType;  
    resourcesDetails = KBresourcesDetails;  
    cus = pcus;  
    cus->addResources(this);  
}
```

//Support ticket class Methods Implementation

```
void SupportTicket::displaySTdetails() {  
    cout << "Ticket ID: " << ticketID << endl;  
    cout << "Ticket Type: " << ticketType << endl;  
    cout << "Date: " << date << endl;  
    cout << endl << "*****" << endl;  
}
```

```
SupportTicket::~SupportTicket()
```

```
{  
    cout << "Deleting the Support ticket.." << endl ;  
}
```

```
SupportTicket::SupportTicket() {
```

```
    ticketID = "";  
    ticketType = "";  
    date = "";  
}
```

```
SupportTicket::SupportTicket(string tid, string ttype, string sdate, SupportAgent* csAgent) {
```

```
    ticketID = tid;  
    ticketType = ttype;  
    date = sdate;  
    csAgent->addSupportTicket(this);  
}
```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

Implementation of the main Method

```
int main()
{
    //Creating a Agent object
    string aUserName = "Amal";\
    string userEmail = "yajas@gmail.com";
    string agentID = "A123";

    SupportAgent* agent = new SupportAgent(&aUserName, &userEmail, &agentID);

    //Creating Support ticket object
    string sTID = "001";
    string sTType = "Technical";
    string sTDate = "2023-06-13";

    SupportTicket* ticket1 = new SupportTicket(sTID, sTType, sTDate , agent);

    //Creating Service plan object
    ServicePlan* splan1 = new ServicePlan("Standered", 1000);

    //Creating a RegisteredCustomer object
    string userName = "Yajas";
    string rcustomerID = "C123";
    string accountUserName = "Yajas";
    string accountEmail = "yajas@gmail.com";
    string accountPassword = "yasa2123";
    RegisteredCustomer *u = new RegisteredCustomer(&userName, &userEmail ,
                                                    &rcustomerID, accountUserName,
                                                    accountEmail, accountPassword , splan1);

    // Display RegisteredCustomer details

    u->displayDetails();
    u->displayRCustomerDetails();
    u->displayAccount();

    // Display Agent details
    agent->displayDetails();
    agent->displayAgentDetails();

    // Display Support ticket details
    ticket1->displaySTdetails();

    // Creating a Payment object
    string paymentID = "P456";
    double totalAmount = 10000.0;
    string paymentDate = "2023-06-08";
    Payment payment(paymentID, totalAmount, paymentDate);
}
```


BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

```
// Display payment details
payment.displayPayDetails(u);

//Creating the Feedback Objects
Feedback* F1 = new Feedback();
Feedback* F2 = new Feedback();

//Creating the SupportResources Objects
SupportResources* SR1 = new SupportResources();
SupportResources* SR2 = new SupportResources();

// Display Support ticket details
ticket1->displaySTdetails();

//deleting the objects
delete agent;
delete u;
delete F1;
delete F2;
delete SR1;
delete SR2;

return 0;
}
```

BSc (Hons) in Information Technology

Object Oriented Concepts – IT1050

Assignment 2

Student Number	Student Name	Individual Contribution
IT22320582	Jayasundara D.W.S	<ul style="list-style-type: none"> • CRC Cards: Support agent, Payment • Class diagram: Support agent, Payment • C++ Code: Support agent, Payment • UML Notation: Support agent, Payment • Report
IT22305350	Rajapakshe P.H.Y.L	<ul style="list-style-type: none"> • CRC Cards: Account, User, Registered customer • Class diagram: Account, User, Registered customer • C++ Code: Account, User, Registered customer • UML Notation: Account, User, Registered customer
IT22341136	M.A.S Gunathilaka	<ul style="list-style-type: none"> • CRC Cards: Support ticket, Service plan • Class diagram: Support ticket, Service plan • C++ Code: Support ticket, Service plan • UML Notation: Support ticket, Service plan
IT22315632	R.N.D.K.Rajapaksha	<ul style="list-style-type: none"> • CRC Cards: Feedback, Support resources • Class diagram: Feedback, Support resources • C++ Code: Feedback, Support resources • UML Notation: Feedback, Support resources
IT22337412	D.Rajapaksha	<ul style="list-style-type: none"> • CRC Cards: Not participated. • Class diagram: Not participated. • C++ Code: Not participated. • UML Notation: Not participated.