



Topic : Hotel Reservation System for Tourists

Group no : MLB_8.2_01

Campus : Malabe

Submission Date : 20/05/2022

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
IT21266782	Rathnayake D.T	077 186 7454
IT21253294	Rathnayake M.R.T.N	0760457872
IT21480522	Muthukumari T.R.V	077 379 2997
IT21263576	Polhengoda P.M.Y.Y.B	0767801452
IT21253980	Perera P.W.C.L	077 821 7585

Exercise 1:

1) User Requirements

- A guest can choose the hotel from preferred location and check the availability of rooms
- The guest can view the packages of the hotel.
- The guest in an online Hotel Reservation system needs to register providing details.
- Then the Registered user can log in to the system using his/her credentials.
- A registered user can requests available offers and selects a package
- The Registered user enters payment details to reserve rooms.
- The system validates the payment and generates the invoice.
- The Registered User views the invoice according to the reservation.
- The Registered User can view the status of the reservation details.
- The registered user can sends feedbacks to the system.
- The admin log in to the system using his/her credentials.
- The admin can manages the offers.
- The admin checks the reservation details and update the rooms.
- Admin creates reports for the hotel reservation system.

2) Noun & verb Analysis

(Noun are in yellow and Verb are in blue),

The guest visits to the hotel reservation system to reserve rooms. The guest can choose the hotel from preferred location. The user can view the packages. Then user selects the room type and checks the availability of the rooms. If all of the rooms are booked in the selected hotel then user has to choose another hotel or exit from the system. For the reservation of rooms, the user has to register to the system by providing full name, age, email, passport number, mobile number. Then the Registered customer can log in to the system using his/her credentials. A registered customer can requests available offers and select a package. The registered customer enter the payment details to reserve rooms by selecting a payment type (visa/ MasterCard /credit card). Then the system validate the payment and generates the invoice. The Registered customer views the invoice according to the reservation. Then the member can view the status of the reservation. The registered customer can also send feedbacks to the system for their service. The admin log in to the system using his/her credentials. He/she can manages the offers. Also the admin checks the reservation details and update the availability of the rooms. Finally, the admin creates an annual report for the hotel reservation system.

3) Identified classes using Noun Analysis,

Identified classes;

class

- Guest
- Room
- Hotel
- Location
- packages
- Registered customer
- Offers
- Payment
- Invoice
- Reservation
- Feedback
- Admin
- Report
- Creditcard
- Master card
- Visacard

Reasons for Rejecting nouns

1. Redundancy

- User-refers to the same person “guest”
- Payment details-refers to the “payment”
- Member-refers to the same person “registered customer”
- Reservation details-refers to the “reservation”

2. Attribute

3. Room type

4. Fullname,age,email,passport number,mobile number

5. Credentials

6. payment type(visa,master card,credit card)

3. Outside the scope

- hotel reservation system

Exercise 2:

CRC Cards for the hotel reservation system for tourists

Guest	
<u>Responsibility</u>	<u>Collaborators</u>
Selected hotel	Hotel
Choose preferred location	Location
Check available rooms	Room
View packages	Packages

Room	
<u>Responsibility</u>	<u>Collaborators</u>
Provide room details	
Reserve rooms	Reservation

Hotel	
<u>Responsibility</u>	<u>Collaborators</u>
Choose preferred hotel	Registered Customer
Store hotel details	

Payment	
<u>Responsibility</u>	<u>Collaborators</u>
Calculate and store amount of deposit	
Validate	

Credit card	
<u>Responsibility</u>	<u>Collaborators</u>
Store card details	payment

Master card	
<u>Responsibility</u>	<u>Collaborators</u>
Store card details	payment

Visa card	
<u>Responsibility</u>	<u>Collaborators</u>
Store card details	payment

Location	
<u>Responsibilities</u>	<u>Collaborators</u>
Location details	
Update locations	

Packages	
<u>Responsibilities</u>	<u>Collaborators</u>
Package details	
Update packages	

Invoice	
<u>Responsibilities</u>	<u>Collaborators</u>
store invoice details	
Generate invoice details	Offers

Reservation	
<u>Responsibilities</u>	<u>Collaborators</u>
Set reservation details	
Update reservation details	
Confirm reservation	Registered customer

Offers	
<u>Responsibilities</u>	<u>Collaborators</u>
Select offers	Registered customer
Update offers	Admin
Store offer details	

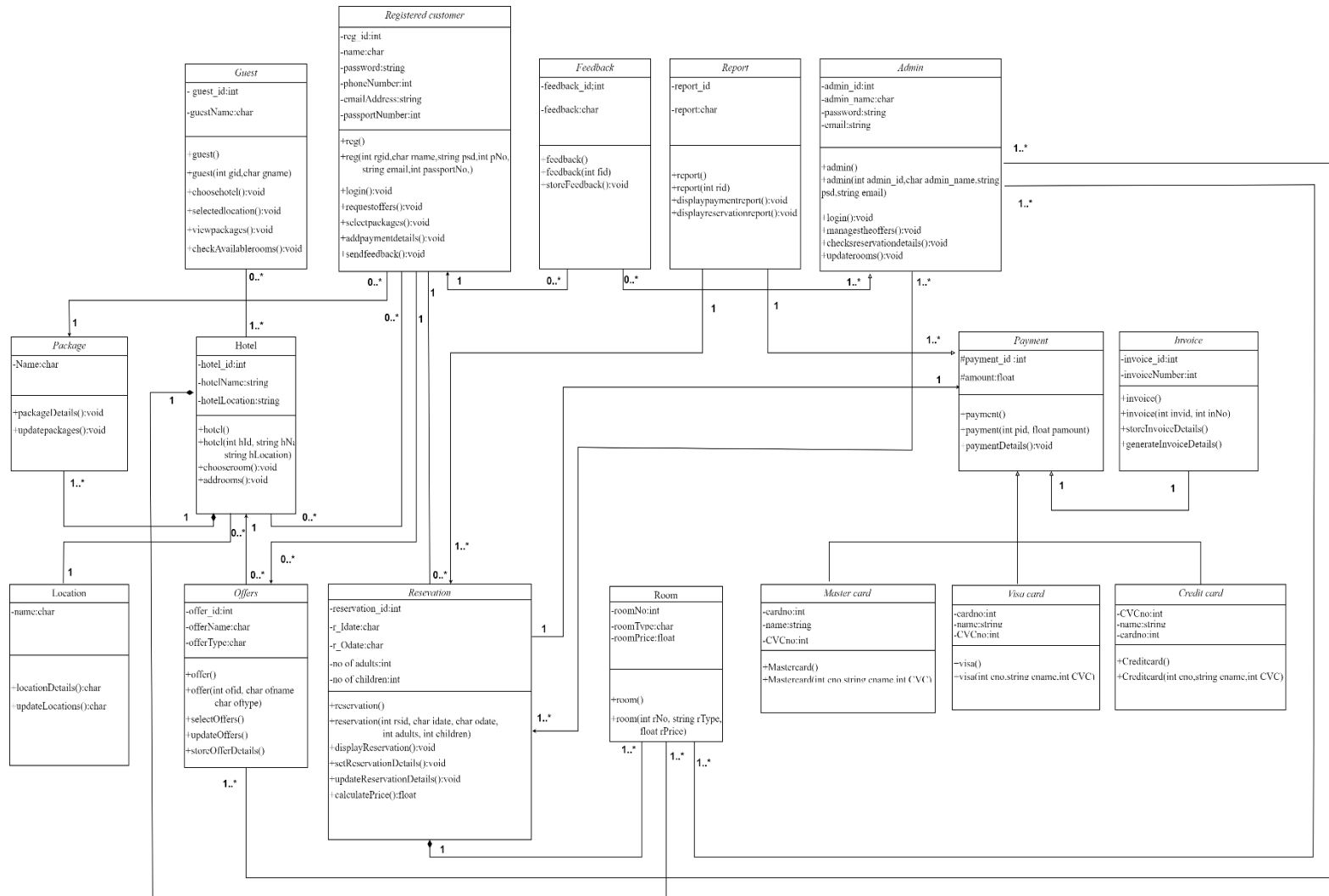
Admin	
<u>Responsibilities</u>	<u>Collaborators</u>
Log into the system	
Manages the offers	Offers
Checks the reservation details	Reservation
Update rooms	Room
Reply to customer feedback	Feedback

Feedback	
<u>Responsibility</u>	<u>Collaborators</u>
Store customer details	

Report	
<u>Responsibility</u>	<u>Collaborators</u>
List of reservation report	Reservation
Display payment report	Payment
Annual report	

Exercise 3:

Class diagram for the hotel reservation system for tourists



Exercise 4:

C++ Coding

```
#include<iostream>
```

```
#include<string>
```

```
using namespace std;
```

```
//class decleration
```

```
class guest
```

```
{
```

```
private:
```

```
int guestId;
```

```
char guestName;
```

```
public:
```

```
guest();
```

```
guest(int gid, string gname);
```

```
void choosehotel();
```

```
void selectlocation();
```

```
void viewpackages();
```

```
void checkAvailablerooms();
```

```
};
```

```
class room
{
private:
int roomNo;
string roomType;
float roomPrice;
public:
room();
room(int rNo, string rType, float rPrice);
};

class hotel
{
private:
int hotelId;
string hotelName;
string hotelLocation;
room *rooms[SIZE];
public:
hotel();
hotel(int hId, string hName, string hLocation);
void chooseroom();
void addrooms(room *r1, room *r2);
```

```
};
```

```
class payment
```

```
{
```

```
protected:
```

```
int paymentid;
```

```
float amount;
```

```
public:
```

```
payment();
```

```
payment(int pid, float pamount);
```

```
void paymentdetails();
```

```
};
```

```
class visa :public payment {
```

```
private:
```

```
int cardno;
```

```
string name;
```

```
int cvcno;
```

```
public:
```

```
visa();
```

```
visa(int cno, string cname,int cvc);
```

```
};
```

```
class mastercard :public payment {
```

```
private:
```

```
int cardno;
```

```
string name;
```

```
int cvcno;
```

```
public:
```

```
mastercard();
```

```
mastercard(int cno, string cname,int cvc);
```

```
};
```

```
class creditcard :public payment {
```

```
private:
```

```
int cardno;

string name;

int cvcno;


public:

creditcard();

creditcard(int cno, string cname,int cvc);


};
```

```
class registered_user

{

private:

int r_id;

char name;

string password;

string emailAddress;

int passportno;

packages *package;

offers *offer;
```

```
public:
registered_user();
registered_user(int reg_id,char rName,string rPassword,string
rEmailaddress,int rPassportno);
void log in();
    void requests offers(offers *o);
void select packages(packages *p);
void payment details();
void view invoice();
void send feedback();
};
```

```
class locations
```

```
{
```

```
private:
```

```
int lId;
```

```
char locationname;
```

```
public:
```

```
locations();
```

```
locations(int loc_id,char lname);
```

```
void location details();
```

```
};
```



```
class packages
{
private:
char packagename;
int pId;

public:
packages();
packages(int packageid,char pname);

};
```

```
class reservation
{
private:
int reservation_id, noofadults, noofchildren;
char r_Idate, r_Odate;
room *rooms[SIZE];

public:
```

```
reservation();  
reservation(int rsid, char idate, char odate,int adults, int children);  
void displayReservation();  
void setReservationDetails();  
void updateReservationDetails();  
void addrooms();  
float calculatePrice();  
  
};
```

```
class invoice {  
  
    private:  
  
    int invoice_id;  
    int invoiceNumber;  
    payment *payments;
```

public:

invoice();

invoice(int invid, int inNo, payment *p);

void storeInvoiceDetails();

void generateInvoiceDetails();

};

class offers {

private:

int offer_id;

char offerName;

char offerType;

hotel *hotels;

public:

offer();

offer(int ofid, char ofname, char oftype, hotel *h);

```

    void selectOffers();

    void updateOffers();

    void storeOfferDetails();

};

class admin
{
    private:

        int adminId;

        char adminName[30];

        char password[20];
char email[30];

    public:

        admin();

        admin(int adminId,char adminName[],char password,char
email[]);

        void login();

        void displayAdminDetails();

        void managestheoffers();

        void checksreservationdetails();

        void updaterooms();

```

```
};
```

```
class feedback
```

```
{
```

```
    private:
```

```
        int feedbackId;
```

```
        char feedback[100];
```

```
    public:
```

```
        feedback();
```

```
        feedback(int fid,char fdb[]);
```

```
        void storeFeedback();
```

```
};
```

```
class report
```

```
{
```

```
    private:
```

```
        int reportId;
```

```
        char report[100];
```

```
    public:
```

```
    report();  
    report(int rid,char report);  
    void displaypaymentreport();  
    void displayreservationreport();  
};
```

//function implementation

```
guest::guest(){ }  
guest::guest(int gid, string gname)  
{  
    guestid = gid;  
    guestName = gname;  
}  
void guest::choosehotel(){ }  
void guest::selectlocation(){ }  
void guest::viewpackages(){ }  
void guest::checkAvailablerooms(){ }  
  
room::room(){ }  
room::room(int rno, sting rtype, string rPrice)  
{
```

```
roomNo = rno;
roomType = rtype;
roomPrice = rPrice;
}
room::(){}
room::(int rNo, string rType, float rPrice){ }
```

```
hotel::hotel(){}
hotel::hotel(int hId, string hName, string hLocation)
{
    hotelId = hId;
    hotelName = hName;
    hotelLocation = hLocation;
}
void hotel::chooseroom(){}
void hotel::addrooms(room *r1, room *r2){ }
```

```
payment::payment(){}

payment::payment(int pid, float pamount)
{
    paymentid = pid;
```

```
amount = pamount;
```

```
}
```

```
void payment::paymentdetails(){ }
```

```
visa::visa() { }
```

```
visa::visa(int cno, string cname, int cvc)
```

```
{
```

```
cardno = cno;
```

```
name = cname;
```

```
cvcno = cvc;
```

```
}
```

```
mastercard::mastercard(){ }
```

```
mastercard::mastercard(int cno, string cname, int cvc)
```

```
{
```

```
cardno = cno;
```

```
name = cname;
```

```
cvcno = cvc;
```

```
}
```



```

creditcard::creditcard(){ }

creditcard::creditcard(int cno, string cname, int cvc)
{
    cardno = cno;
    name = cname;
    cvcno = cvc;

}

void package details();
void update deatails();
}

//Class implementation

registered_user::registered_user(){
}

registered_user::registered_user(int reg_id,char rName,string
rPassword,string rEmailaddress,int rPassportno)
{
    r_id=reg_id;
    name=rName;
    password=rPassword;
    emailAdress=rEmailaddress;
    passportno=rPassportno;
}

```

```

}
void registered_user::log in(){
}
void registered_user::requests offers(offers *o){
}
void registered_user::select packages(packages *p){
}
void registered_user::payment details(){
}
void registered_user::view invoice(){
}
void registered_user::send feedback(){
}

```

```

location::locations(){
}
location::locations(int locid ,char lname){
    lId=loc_id;
    locationname=lname;
}
void location::location details(){
}

```

```
packages::packages(){  
}  
packages::packages(int packageid,char pname){  
    packagename=pname;  
    pId=packageid;  
}  
void packages::package details(){  
}  
void packages::update deatails(){  
}
```

```
reservation::reservation(){ }  
reservation::reservation(int rsid, char idate, char odate,int adults, int  
children)  
{  
    reservation_id = rsid;  
    reservationInDate = idate;  
    reservationOutDate = odate;  
    numberOfAdults = adults;  
    numberOfChildren = children;  
}  
void reservation::displayReservation(){ }
```

```
void reservation::setReservationDetails(){ }  
void reservation::updateReservationDetails(){ }  
void reservation::addrooms(){ }  
float reservation::calculatePrice(){ }
```

```
invoice::invoice() { }  
invoice::invoice(int invid, int inNo, payment *p)  
{  
    invoice_id = invid;  
    invoiceNumber = inNo;  
    payment *payments = payment *p;  
}  
void invoice::storeInvoiceDetails(){ }  
void invoice::generateInvoiceDetails(){ }
```

```
offers::offers(){ }  
offers::offer(int ofid, char ofname, char oftype, hotel *h)  
{  
    offer_id = ofid;  
    offerName = ofname;  
    offerType = oftype;  
    hotel *hotels = hotel *h;
```

```
}
```

```
void offers::selectOffers(){}
```

```
void offers::updateOffers(){}
```

```
void offers::storeOfferDetails(){}
```

```
admin::admin(int id,char name[],char Password[],char Email[]){
```

```
adminId=id;
```

```
strcpy(adminName,name);
```

```
strcpy(password>Password);
```

```
strcpy(email>Email);
```

```
}
```

```
admin::displayAdminDetails(){
```

```
cout<<"Display Admin ID:"<<adminId<<endl;
```

```
cout<<"Display Admin Name:"<<dminName<<endl;
```

```
cout<<"Display Admin Email:"<<email<<endl;
```

```
}
```

```
feedback::feedback(int id,char fdb[]){
```

```
feedbackId=id;
```

```
strcpy(feedback,fdb);
```

```
}
```

```
report::report(int rid,char report[]){  
reportId=rid;  
strcpy(report,report);  
}
```

```
//main program
```

```
int main()  
{  
registered_user  
r1(1111,"Randuni","tR#hj","randuni@gmail.com",8764);
```

```
location l1(8906,"Colombo");
```

```
packages p1(6524,"package1");
```

```
    payment p1(1, 20000);
```

```
visa v1(4562135498758546, "tharindu", 256);
```

```
mastercard m1(4216875236549825, "nipun", 546);
```

```
creditcard c1(2564879565321256, "anjana", 987);
```

```
guest g1(1, "Sanduni");
```

```
room r1(1, "Single room", "45000");
```

```
hotel h1(1, "ABC Hotel", "Colombo");
```

```
    reservation p1(123, "03/05/2022", "06/05/2022", 2, 3);
```

```
invoice v1(123456, 200);

offers m1(123456789, "christmas", "season offer", "ABC Hotel");

admin *admin1= new
admin(1234,'Chaith','KH@','charith@gmail.com');

feedback *f1 = new feedback(1,'good');

reprot *r1 = new report(11,'gasdginasgid');

admin1->displayAdminDetails();

return 0;

}
```