Topic: Airline Ticket Reservation

Group no: MLB_10.02_09

Campus: Malabe

Submission Date: 2022.05.17

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
IT21381218	Perera L.P.S.R	071 4777848
IT21203794	M.P.Chameen Shamika	077 8323347
IT21388620	P.V.I.R. Samanayaka	076 0557661
IT21384738	Ranasinghe R.M.P.A	071 9304362
IT21226878	P. Kavindu Naveen Premathilake	071 3702074

Description

The airline ticket reservation system is an online air ticket booking system that allows to customers to book their flight tickets from anywhere in the world, at any time. To any possible destination. Unregistered customers and registered customers may also visit the page to display the data, but unregistered customers must register to system and validated by the system in order to receive further services.

A customer should register in the system to visit the data displayed and to receive the further services provided. For registering a customer should fill the given application at the beginning customer must enter his first name, last name, Email Address, Phone number, date of birth, country and submit the same as per the instructions. Then it will be validated and be conformed through the email. After that the user should enter their username and password to the enter the system. Then the user can direct to the home page. After that user can get more services, such as booking tickets, booking hotels, checking payment details and user can edit data id they need to do so. Then the registered person can access the page through the link provided at the home page and can choose the destination and flight details and also hotel booking. The registered customer can choose the destination, the available flight & time and date, enter them in the system after all that is done customer direct to the payment page. After the payment customer will receive a generated email with all detail.

Admins add the all the flight details like wise (flight number, flight type, destination, number of seat that available, class of the seat, transit details) and also hotel details. Only Admins will be charge of adding new flight details, removing existing flight details and adding and removing hotel details.

Requirements

1. Any visitor can register as a customer according to available registering plans.
2. Any user can login to the system by providing their ID and password.
3. If a customer has authenticated, he or she can display, delete, or update his or her profile information.
4. Customer can search flights and he/she can select the date, package, seat and hotel as he/she want.
5. Customer can book a flight.
6. If a customer wants to book a flight, customer must check the availability of the flight for the particular date from the system.
7. Once the system notifies that the flight is available, customer can book the flight.
8. Customer can do payments can be made using Credit/Debit card.
9. After the payment, the system sends the ticket details to the customer's email address.
10. If customer wants to the cancel the booking, there is an option to cancel it.
11. Airline staff must first log into the system before beginning the tasks.
12. Airline staff can generate reports, check flight availability, check and manage customer details.
13. Before doing the tasks, the admin also must first log into the system.
14. Admin can check the reports, Manage customer and airline staff user Accounts.
15. Admin can add hotels to the system and manage them.

Noun/ Verb Analysis

- **Noun**
- **Verb**

1. Any visitor can register as a customer according to available registering plans.
2. Any user can login to the system by providing their ID and password.
3. If a customer has authenticated, he or she can display, delete, or update his or her profile information.
4. Customer can search flights and he/she can select the date, package, seat and hotel as he/she want.
5. Customer can book a flight.
6. If a customer wants to book a flight, customer must check the availability of the flight for the particular date from the system.
7. Once the system notifies that the flight is available, customer can book the flight.
8. Customer can do payment can be made using Credit/Debit card.
9. After the payment, the system sends the ticket details to the customer's email address.
10. If customer wants to the cancel the booking, there is an option to cancel it.
11. Airline staff must first log into the system before performing the tasks.
12. Airline staff can generate reports, check flight availability, check and manage customer details.
13. Before doing the tasks, the admin also must first log into the system.
14. Admin can check the reports, Manage customer and airline staff user Accounts.
15. Admin can add hotels to the system and manage them.

Identifying classes using noun/ verb analysis

Visitor	Outside Scope Of System
Customer	Class
Plans	Redundant
User	Class
System	Outside Scope Of System
ID	Attribute
Password	Attribute
Profile Information	Meta Language
Flights	Class
Date	Attribute
Package	Class
Seat	Attribute
Hotel	Class
Payment	Class
Credit/Debit Card.	Attribute
Ticket	Class
Email Address	Attribute
Airline Staff	Class
Reports	Class
Flight Availability	Meta Language
Admin	Class
Accounts.	Redundant

CRC Card

Customer	
Responsibilities	**Collaborations**
Search flights	Flight
Select package	Package
Select hotel	Hotel
Display, delete, or update profile information.	
Cancel ticket	Ticket
Make payment	Payment

Flight	
Responsibilities	**Collaborations**
Store flight number	
Store seat details	

Hotel	
Responsibilities	**Collaborations**
Store hotel registration number	
Store hotel details	

Payment	
Responsibilities	**Collaborations**
Update transaction details	

Package	
Responsibilities	**Collaborations**
Store package details	

Ticket	
Responsibilities	**Collaborations**
Store customer details	Customer
Store flight details	Flight
Store package details	Package

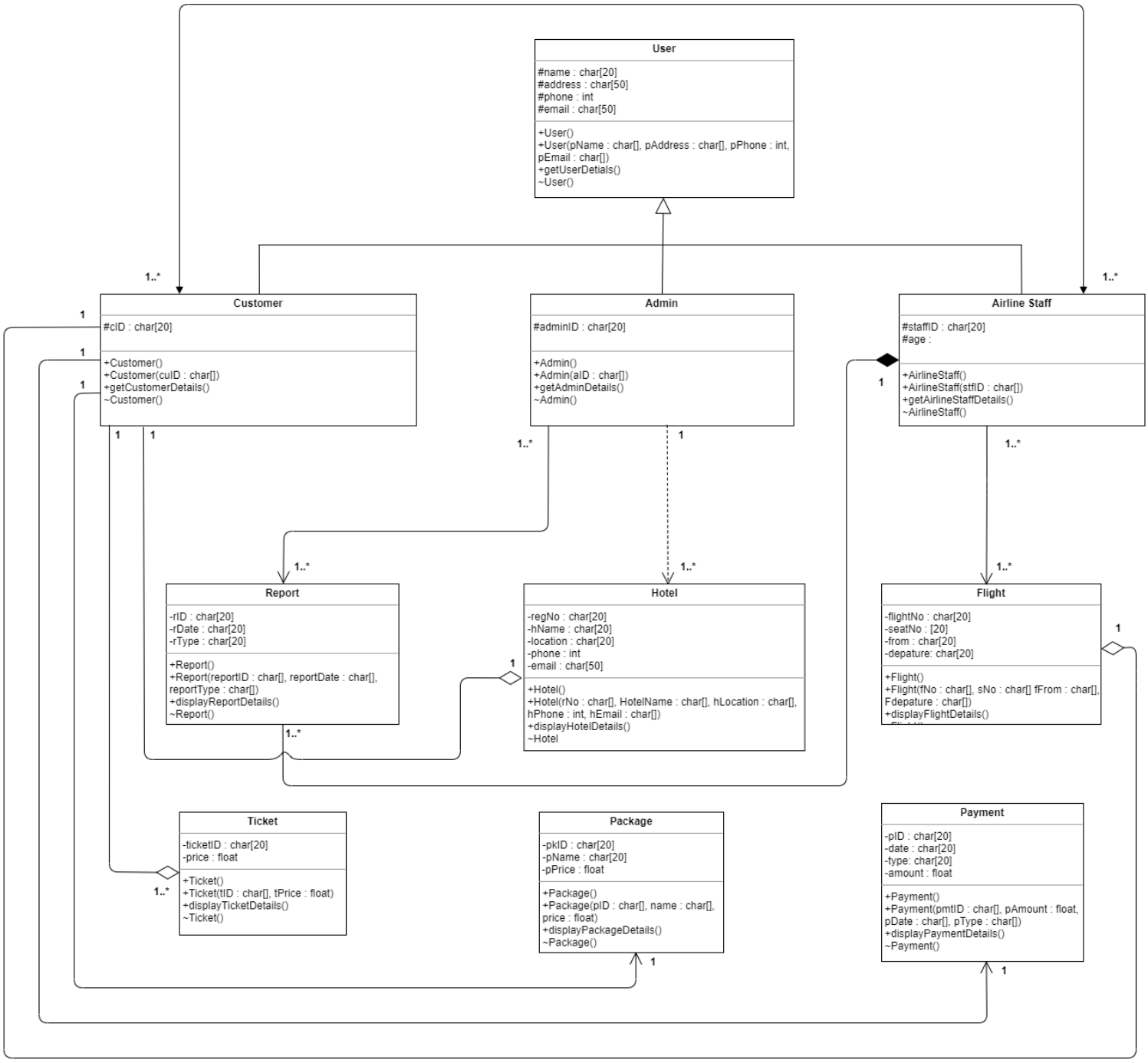
Admin	
Responsibilities	**Collaborations**
Login to the system	
Manage airline staff and customer	Airline staff, Customer
Check reports	Report
Add hotels	Hotel
Add/remove/edit flight details	Flight

Airline staff	
Responsibilities	**Collaborations**
Login to the system	
Manage customer	Customer
Generate reports	Report
Check flight schedule	Flight

Report	
Responsibilities	**Collaborations**
Keep record details	

User	
Responsibilities	**Collaborations**
Keep user details	

UML Diagram


User
#name : char[20]
#address : char[50]
#phone : int
#email : char[50]
+User()
+User(pName : char[], pAddress : char[], pPhone : int, pEmail : char[])
+getUserDetials()
~User()
Customer
#cID : char[20]
+Customer()
+Customer(cuID : char[])
+getCustomerDetails()
~Customer()
Admin
#adminID : char[20]
+Admin()
+Admin(aID : char[])
+getAdminDetails()
~Admin()
Airline Staff
#staffID : char[20]
#age :
+AirlineStaff()
+AirlineStaff(stfID : char[])
+getAirlineStaffDetails()
~AirlineStaff()
Report
-rID : char[20]
-rDate : char[20]
-rType : char[20]
+Report()
+Report(reportID : char[], reportDate : char[], reportType : char[])
+displayReportDetails()
~Report()
Hotel
-regNo : char[20]
-hName : char[20]
-location : char[20]
-phone : int
-email : char[50]
+Hotel()
+Hotel(rNo : char[], HotelName : char[], hLocation : char[], hPhone : int, hEmail : char[])
+displayHotelDetails()
~Hotel
Flight
-flightNo : char[20]
-seatNo : [20]
-from : char[20]
-depature: char[20]
+Flight()
+Flight(fNo : char[], sNo : char[] fFrom : char[], Fdepature : char[])
+displayFlightDetails()
Ticket
-ticketID : char[20]
-price : float
+Ticket()
+Ticket(tID : char[], tPrice : float)
+displayTicketDetails()
~Ticket()
Package
-pkID : char[20]
-pName : char[20]
-pPrice : float
+Package()
+Package(pID : char[], name : char[], price : float)
+displayPackageDetails()
~Package()
Payment
-pID : char[20]
-date : char[20]
-type: char[20]
-amount : float
+Payment()
+Payment(pmtID : char[], pAmount : float, pDate : char[], pType : char[])
+displayPaymentDetails()
~Payment()

C++ Code

```
#include<iostream>
#include<cstring>
using namespace std;

//User class
class User{
	protected:
		char name[20];
		char address[50];
		char email[50];
		int phone;
	public:
		User(){
			strcpy(name,"");
			strcpy(address,"");
			strcpy(email,"");
			phone = 0;
		}
		User(char pName[], char pAddress[], char pEmail[], int pPhone){
			strcpy(name,pName);
			strcpy(address,pAddress);
			strcpy(email,pEmail);
			phone = pPhone;
		}
		void getUserDetails(){
			cout<<"Enter Name: "<<name<<endl;
			cout<<"Enter Address: "<<address<<endl;
			cout<<"Enter Email: "<<email<<endl;
```

```
                cout<<"Enter phone number: "<<phone<<endl;
        }
};


//Customer class
class Customer: public User{
    protected:
        char cID[20];
    public:
       Customer(){
        strcpy(cID,"");
        }
        Customer(char pName[], char pAddress[], char pEmail[], int pPhone, char
cuID[]):User
        (pName, pAddress, pEmail, pPhone){
            strcpy(cID,cuID);
        }
        void getCustomerDetails(){
            cout<<"Customer ID: "<<cID<<endl;
        }
};


//Admin class
class Admin: public User{
    protected:
        char adminID[20];
    public:
       Admin(){
```

```
            strcpy(adminID,"");
        }
        Admin(char pName[], char pAddress[], char pEmail[], int pPhone, char
aID[]):User
        (pName, pAddress, pEmail, pPhone){
            strcpy(adminID,aID);
        }
        void getAdminDetails(){
            cout<<"Admin ID: "<<adminID<<endl;
        }
};

//Airline_Staff class
class Airline_Staff: public User{
    protected:
        char staffID[20];
    public:
        Airline_Staff(){
            strcpy(staffID,"");
        }
        Airline_Staff(char pName[], char pAddress[], char pEmail[], int pPhone, char
stfID[]):User
        (pName, pAddress, pEmail, pPhone){
            strcpy(staffID,stfID);
        }
        void getAirline_StaffDetail(){
            cout<<"Airline Staff ID: "<<staffID<<endl;
        }
};
```

```
//Report class
class Report{
	private:
		char rID[20];
		char rDate[20];
		char rTyoe[20];
	public:
	   Report(){
		strcpy(rID,"");
		strcpy(rDate,"");
		strcpy(rTyoe,"");
		}
		Report(const char reportNo[],const char reportDtae[],const char
reportType[]){
			strcpy(rID,reportNo);
			strcpy(rDate,reportDtae);
			strcpy(rTyoe,reportType);
		}
		void displayReportDetails(){
			cout<<"Report No: "<<rID<<endl;
			cout<<"Report date: "<<rDate<<endl;
			cout<<"Report type: "<<rTyoe<<endl;
		}
		~Report(){};
};
```

```
//Hotel class
class Hotel{
	private:
		char regNo[20];
		char hName[20];
		char location[20];
		int phone;
		char email[50];
	public:
	   Hotel(){
		strcpy(regNo,"");
		strcpy(hName,"");
		strcpy(location,"");
		phone= 0;
		strcpy(email,"");
		}
		Hotel(const char rNo[],const char hotelName[],const char hLocation[], int
hPhone,const char hEmail[]){
			strcpy(regNo,rNo);
			strcpy(hName,hotelName);
			strcpy(location,hLocation);
			phone=hPhone;
			strcpy(email,hEmail);
		}

		void displayHotelDetails(){
			cout<<"Hotel registration no: "<<regNo<<endl;
			cout<<"Hotel name: "<<hName<<endl;
			cout<<"Hotel loction: "<<location<<endl;
			cout<<"Hotel phone: "<<phone<<endl;
```

```
                cout<<"Hotel email: "<<email<<endl;
        }
        ~Hotel(){};
};


//Flight class
class Flight{
    private:
        char flightNo[20];
        char seatNo[10];
        char from[20];
        char depature[20];
    public:
       Flight(){
        strcpy(flightNo,"");
        strcpy(seatNo,"");
        strcpy(from,"");
        strcpy(depature,"");
        }
        Flight(const char fNo[], const char sNo[], const char fFrom[], const char
fdepature[]){
           strcpy(flightNo,fNo);
           strcpy(seatNo,sNo);
           strcpy(from,fFrom);
           strcpy(depature,fdepature);
        }
        void displayFlightDetails(){
            cout<<"Flight no: "<<flightNo<<endl;
            cout<<"Flight seat no: "<<seatNo<<endl;
```

```
                cout<<"Flight from: "<<from<<endl;
                cout<<"Flight depature: "<<depature<<endl;
        }
        ~Flight(){};
};


//Ticket class
class Ticket{
    private:
       char ticketID[20];
       float price;
    public:
       Ticket(){
            strcpy(ticketID,"");
            price= 0;
            }
            Ticket(const char tID[], float tPrice){
                strcpy(ticketID,tID);
                price = tPrice;
            }
            void displayTicketDetails(){
                cout<<"Ticket no: "<<ticketID<<endl;
                cout<<"Ticket price: Rs."<<price<<endl;
            }
            ~Ticket(){};
};
```

```
//Package class
class Package{
	private:
		char pkID[20];
		char pName[20];
		float pPrice;
	public:
		Package(){
			strcpy(pkID,"");
			strcpy(pName,"");
			pPrice = 0;
		}
	Package(const char pID[], const char name[], float price){
	  strcpy(pkID,pID);
	  strcpy(pName,name);
	  pPrice = price;
		}
		void displayPackageDetails(){
			cout<<"Package ID: "<<pkID<<endl;
			cout<<"Package name: "<<pName<<endl;
			cout<<"Package price: Rs."<<pPrice<<endl;
		}
		~Package(){};
};


//Payment class
class Payment{
	private:
```

```
        char pID[20];
        char date[20];
        char type[20];
        float amount;
    public:
      Payment(){
        strcpy(pID,"");
        strcpy(date,"");
        strcpy(type,"");
        amount = 0;
        }
        Payment(const char pmtID[], const char pDate[], const char pType[], float
pAmount){
          strcpy(pID,pmtID);
        strcpy(date,pDate);
        strcpy(type,pType);
        amount = pAmount;
         }
        void displayPaymentDetails(){
            cout<<"Payment ID: "<<pID<<endl;
            cout<<"Payment date: "<<date<<endl;
            cout<<"Payment type: "<<type<<endl;
            cout<<"Payment amount: Rs."<<amount<<endl;
        }
        ~Payment(){};
};
```

```
//Main Program
int main(){
	Customer c1;
	Admin a1;
	Airline_Staff stf1;
	Report r1;
	Hotel h1;
	Flight f1;
	Ticket t1;
	Package pk1;
	Payment pmt1;

	c1.getCustomerDetails();
	a1.getAdminDetails();
	stf1.getAirline_StaffDetail();
	r1.displayReportDetails();
	h1.displayHotelDetails();
	f1.displayFlightDetails();
	t1.displayTicketDetails();
	pk1.displayPackageDetails();
	pmt1.displayPaymentDetails();

	return 0;
}
```