

Air Ticket Reservation



Session 2023-2027

Submitted by:

Huzaiifa Arshad 2023-CS-86

Supervised By:

Dr. Muhammad Awais Hassan

Course:

CSC-102 Programming Fundamentals

Department of Computer Science

University of Engineering and Technology

Lahore Pakistan

Table of Content

- **Overview**
- **Users of Application**
 - **Admin**
 - **User**
- **Admin Capabilities**
- **Functionalities**
- **Wireframes**
- **Data Variables**
 - **Array**
 - **Basic/main data structures**
 - **Passenger data structures**
 - **Admin Data structures**
- **Functions**
 - **Function Prototypes**
 - **Function Working Flow**
- **FlowChart**
- **Complete Code of the Business Application**
- **Weakness in the Business Application**
- **Future Directions**

- **Overview**

Air Ticket Reservation is basically about the ticket reservation of the passenger and the admin manages its data. Admin also manages the flights. Passenger can enter data and then reserve the ticket if flight is available. I have tried my best to put all my efforts through implementing all possible functionalities in this system.

- **Users of Application**

There are two users in my application.

1. Admin
2. User

- **Admin**

My first user is admin or the owner. He has a lot of functionalities. He can add flights so that user can book that flight. He can also view passenger details and travel data. He can view the booked flights.

- **User**

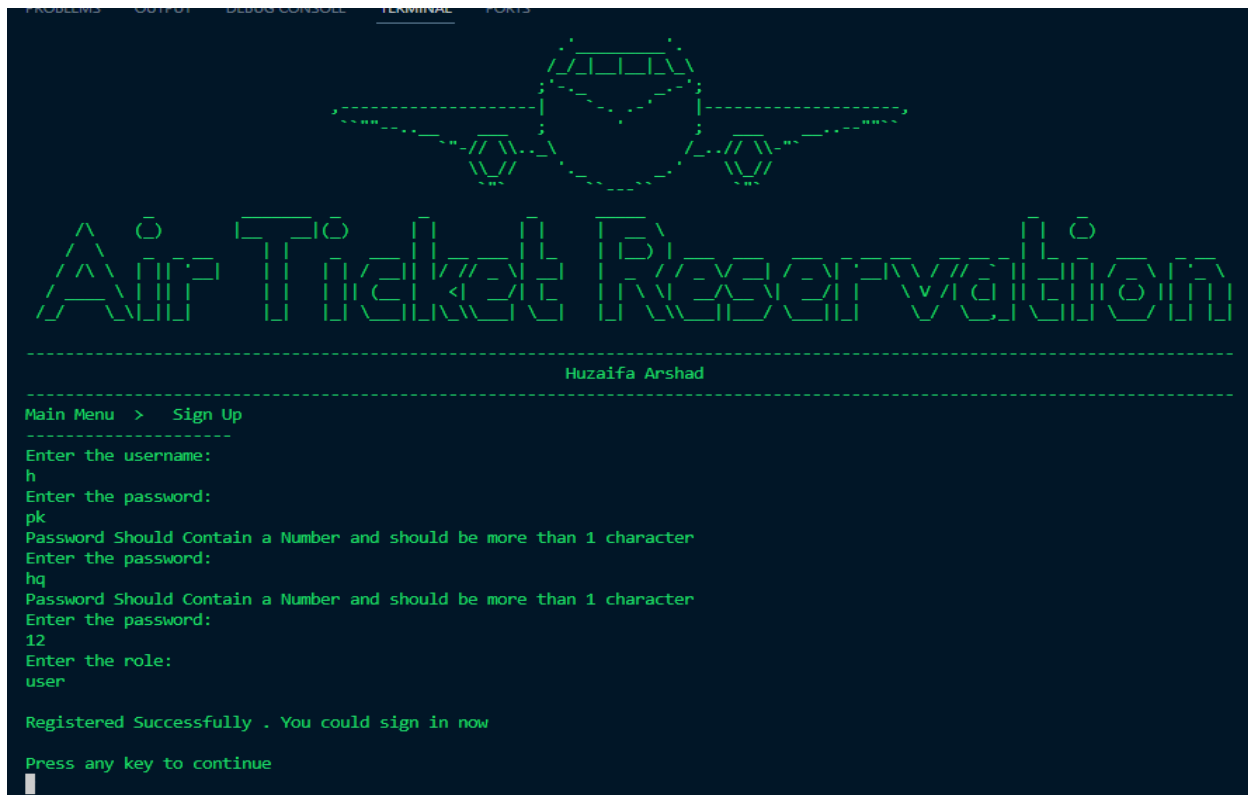
My second user is the Client. He has a lot of functionalities. He can register and book the flight if available. He can cancel the flight and provide a personal feedback and rating. He can also view the booked flight.

- **Admin Capability**

He has a lot of functionalities. He can add flights so that user can book that flight. He can also view passenger details and travel data. He can view the booked flights. He can allot seats. He can see the sorted passengers.

- **Functionalities:**

As a	I want to perform	So that I can
Admin	View Passenger Information	check User
	View Passenger Travel data	Check their details
	Add new flights	able user to book flight
	View added flights	see details
	Allot Seats	Get informed
	Ordered Passengers	Sort out
	View booked flights	calculate number of passengers
Client	Registration of the passenger	book the flight
	Enter Flight details	add details
	Booking Flight If available and get invoice	go for my journey
	View available fights	Get my ticket
	View my Booking	get informed
	Cancel the Flight	cancel it
	Provide Flight Feedback	Check system
	Book to International Countries	Go for visit



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Air Ticket Reservation
-----
Huzaifa Arshad
-----
Main Menu > Sign Up
-----
Enter the username:
h
Enter the password:
pk
Password Should Contain a Number and should be more than 1 character
Enter the password:
hq
Password Should Contain a Number and should be more than 1 character
Enter the password:
12
Enter the role:
user

Registered Successfully . You could sign in now

Press any key to continue
```

Figure.3 Signup Menu

```
-----  
|----- ADMIN MENU -----|  
|-----|  
  
Login Menu > Login > Admin  
-----  
Select one of the following options...  
1- View Passengers Data  
2- View Passengers travel Data  
3- Add new flights  
4- View flights  
5- Allot Seats  
6- view Booked Flights  
7- Ordered Passengers  
8- Admin Stored data  
9- Exit  
Your Option..|
```

Figure.4 Admin Menu


```
1
Login Menu > Login > Admin > Add new flights
-----
Departure from :-
lhr
Arrival To :-
ksk
Trip Type :-
visit
Depart Date :-
12jan
Depart Time :-
12pm
Press any key to continue
```

Figure.5 Add flight

```
2
Login Menu > Login > Admin > View Flights
-----
Departure      Arrival Trip      DepartDate      DepartTime
lhr            ksk      visit      12jan          11:05
lhr            kar      return     13jan          12:30
isb            lhr      visit      15jan          15:30
lhr            ksk      visit      16jan          17:25
lhr            ksk      visit      12jan          12pm
lahore         karachi visit      12jan          12pm
lhr            ksk      visit      12jan          12pm
Press any key to continue
```

Figure.6 view flights


```
-----  
|----- USER MENU -----|  
|                             |  
|                             |  
|-----|  
  
Login Menu > Login > Passenger  
-----  
Select one of the following options...  
1- Registration  
2- Enter Flight Details  
3- Book from available Flights  
4- See booked Flights  
5- Cancel A flight  
6- View Flights  
7- Provide feedback  
8- Booking to Differnt Country  
9- Exit  
Your Option..|
```

Figure.8 User menu

```
Login Menu > Login > Passenger > Registration
-----
Enter your name :-
huz
Enter your CNIC :-
123
Enter your passport no. :-
321
Enter your Phone Number :-
123
Enter your email :-
123
Enter your Gender :-
male
Press any key to continue
█
```

Figure.9 Registration

```
Login Menu > Login > Passenger > Enter Flight Details
-----
Departure from :-
1hr
Arrival To :-
ksk
Trip Type :-
visit
Depart Date :-
12jan
-----Passengers-----
No of Passengers (Adult) :-
2
No of Passengers (Child) :-
2
No of Passengers (Infant) :-
2
-----Class-----
Enter Class :-
business
Press any key to continue
█
```

Figure.10 Enter Flight Details

```

Login Menu > Login > Passenger > Book from available Flights
-----
1- From      1hr  To  ksk  at  12pm      12jan
█
```

Figure.11 Book Flight and invoice

```
Login Menu > Login > Passenger > Book from available Flights
-----
1- From      1hr      To      ksk      at      12pm      12jan
1

Departure    Arrival Trip    Date    Adults    Child    Infants Class
1hr          ksk      visit   12jan    2        2        2        business

-----Expenditures-----
No of Passengers (Adult)(400$) :- 2
No of Passengers (Child) (250$):- 2
No of Passengers (Infant)(0$) :- 2

Your Total Expenditure :- 1500$
Press any key to continue
█
```

Figure.12 Invoice

```
Login Menu > Login > Passenger > Feedback
-----

Please provide feedback for flight SV-734 for passenger Huzaifa:

Comments:
gooh
Rating (1 to 5): 3
Press any key to continue
█
```

Figure.13 User Feedback

```
Thank you for providing feedback!
Flight: SV-734
Passenger: Huzaifa
Your rating: 3Your Feedback is added to our system..

Press any key to continue
```

Figure.14 View Flights

```
Login Menu > Login > Admin > Allot Seats
-----
Departure      Arrival Trip    Date    Adults  Child  Infants Class
              0          0          0
No of Passengers (Adult) :- 0
No of Passengers (Child) :- 0
No of Passengers (Infant) :- 0

Enter Seat Range :- 5
1hr    ksk    visit  12jan  2      2      2      business
No of Passengers (Adult) :- 2
No of Passengers (Child) :- 2
No of Passengers (Infant) :- 2

Enter Seat Range :- 5
Press any key to continue
```

Figure.15

```
Login Menu > Login > Admin > View Passengers travel Data
```

Departure	Arrival	Trip	Date	Adults	Child	Infants	Class
				0	0	0	
lhr	ksk	visit	12jan	2	2	2	business
Lhr	ksk	vist	12JAN	2	2	2	business
lhr	isb	visit	15FEB	3	3	3	economy
isb	kar	return	20JAN	4	4	4	business
lhr	fsb	return	40JAN	7	7	7	economy
lhr	ksk	visit	12jan	12	2	2	business
lhr	ksk	visit	12jan	2	2	2	business
lhr	ksk	visit	12jan	2	2	2	economy
lahore	karachi	visit	12jan	2	2	2	business
lhr	ksk	visit	12jan	2	2	2	business

Press any key to continue

Figure.16

```
Admin Menu > Login > Admin > Ordered Passengers
```

Name	CNIC	Passport	Phone	Email	Gender	Seats	Total
huz	123	321	123	123	male	5	1500
						5	0

Press any key to continue

Figure.17

Login Menu >		Login >		Passenger >		Admin flights		

Departure		Arrival	Trip	DepartDate	DepartTime			
lhr		ksk	visit	12jan	11:05			
lhr		kar	return	13jan	12:30			
isb		lhr	visit	15jan	15:30			
lhr		ksk	visit	16jan	17:25			
lhr		ksk	visit	12jan	12pm			
lahore		karachi	visit	12jan	12pm			
lhr		ksk	visit	12jan	12pm			
Login Menu >		Login >		Admin >		Passengers Data		

Name	CNIC	Passport	Phone	Email	Gender			
huzaifa	42401	3403	0303	Huzaifa12	Male	Male		
ali	4324	5678	0987	Ali13	Male			
hamza	4567	7654	0987	Hamza12	Male			
Talha	4356	3456	3456	Talha12	Male			
huzaifa	424018	0304	0306	huzaifa12	male	male		
huz	234	432	234	huzaifa12	male	male		
huz	234	432	234	234	male			
huz	123	321	123	123	male			
Login Menu >		Login >		Admin >		View Passengers travel Data		

Departure		Arrival	Trip	Date	Adults	Child	Infants	Class
Lhr		ksk	vist	12JAN	2	2	2	business
lhr		isb	visit	15FEB	3	3	3	economy
isb		kar	return	20JAN	4	4	4	business
lhr		fsb	return	40JAN	7	7	7	economy
lhr		ksk	visit	12jan	12	2	2	business
lhr		ksk	visit	12jan	2	2	2	business
lhr		ksk	visit	12jan	2	2	2	economy
lahore		karachi	visit	12jan	2	2	2	business
lhr		ksk	visit	12jan	2	2	2	business
Press any key to continue								

Figure.18

```
_____Welcome to USA Airlines_____
Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. USA - 498
    01-01-2024 8:00AM 10hrs 1000$
2. USA - 658
    09-01-2024 12:00AM 11hrs 1100$
3. USA - 704
    11-01-2024 7:00PM 10hrs 1250$
Select the flight you want to Book:
█
```

Figure.19 Book Flight US

```
You have Sucessfully Booked the flight USA - 658
You can go back to menu and take the ticket

Press 1 to go back to Menu:
█
```

Figure.20 US Flight Booked

```
_____Welcome to British Airlines_____
Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. UK - 233
    09-02-2024 8:00PM 8hrs 800 euro
2. UK - 467
    08-02-2024 12:00AM 11hrs 950 euro
3. UK - 706
    11-01-2024 7:00PM 10hrs 750 euro
Select the flight you want to Book: █
```

Figure.21 Book Flight UK

```
_____Welcome to German Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. GRM - 498
    01-03-2024 08:00AM 10hrs 300 euro
2. GRM - 658
    09-04-2024 12:00AM 11hrs 500 euro
3. GRM - 704
    11-05-2024 07:00PM 10hrs 700 euro
Select the flight you want to Book: 
```

Figure.22 Book Flight German

```
_____Welcome to AUS Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. AUS - 499
    06-05-2024 07:00AM 10hrs 789$
2. AUS - 652
    07-05-2024 11:00AM 11hrs 876$
3. AUS - 728
    13-06-2024 05:00PM 10hrs 675$
Select the flight you want to Book: 
```

Figure.23 Book Flight AUS

```
_____Welcome to AUS Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. AUS - 499
    06-05-2024 07:00AM 10hrs 789$
2. AUS - 652
    07-05-2024 11:00AM 11hrs 876$
3. AUS - 728
    13-06-2024 05:00PM 10hrs 675$
Select the flight you want to Book: █
```

Figure.24 Book Flight US

```
_____Welcome to AUS Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. AUS - 499
    06-05-2024 07:00AM 10hrs 789$
2. AUS - 652
    07-05-2024 11:00AM 11hrs 876$
3. AUS - 728
    13-06-2024 05:00PM 10hrs 675$
Select the flight you want to Book: █
```

Figure.25 Book Flight US

```
_____Welcome to PAK Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. PIA - 445
    04-07-2024 8:00AM 10hrs Rs.50400
2. PIA - 667
    09-01-2024 12:00AM 11hrs Rs.45000
3. PIA - 899
    11-03-2024 7:00PM 10hrs Rs.87000
Select the flight you want to Book: █
```

Figure.26 Book Flight PIA

```
_____Welcome to PAK Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. PIA - 445
    04-07-2024 8:00AM 10hrs Rs.50400
2. PIA - 667
    09-01-2024 12:00AM 11hrs Rs.45000
3. PIA - 899
    11-03-2024 7:00PM 10hrs Rs.87000
Select the flight you want to Book:
3

You have Sucessfully Booked the flight PIA - 899
You can go back to menu and take the ticket

Press 1 to go back to Menu:
█
```

Figure.27 Booked Flight PIA

```
_____Welcome to Qatar Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. QTR - 447
    04-05-2024 08:00AM 10hrs 2509 Riyal
2. QTR - 657
    09-08-2024 12:00AM 11hrs 2550 Riyal
3. QTR - 789
    27-03-2024 07:00PM 10hrs 2540 Riyal
Select the flight you want to Book: █
```

Figure.28 Book Flight Qatar

```
_____Welcome to Turkish Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. TKR - 434
    17-05-2024 08:00AM 10hrs 2700 Liras
2. TKR - 783
    14-08-2024 12:00AM 11hrs 3000 Liras
3. TKR - 985
    03-03-2024 07:00PM 10hrs 4578 Liras
Select the flight you want to Book: █
```

Figure.29 Book Flight Turkey

```
_____Welcome to Dubai Emirates_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. DUB - 474
      17-04-2024 08:00AM 10hrs 2700 Riyals
2. DUB - 703
      14-05-2024 12:00AM 11hrs 3000 Riyals
3. DUB - 905
      03-02-2024 07:00PM 10hrs 4578 Riyals
Select the flight you want to Book: █
```

Figure.30 Book Flight Dubai

```
_____Welcome to Korean Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. KR - 774
    15-04-2024 08:00AM 10hrs 2708 won
2. KR - 103
    15-05-2024 12:00AM 11hrs 3009 won
3. KR - 105
    05-02-2024 07:00PM 10hrs 4571 won
Select the flight you want to Book: █
```

Figure.31 Book Flight Korea

```
_____Welcome to Saudi Airlines_____

Your Comfort is our priority. Enjoy the journey!
Following are the flights

1. SV - 776
    15-04-2024 08:00AM 10hrs 2708 Riyal
2. SV - 163
    15-05-2024 12:00AM 11hrs 3009 Riyal
3. SV - 155
    05-02-2024 07:00PM 10hrs 4571 Riyal
Select the flight you want to Book: █
```

Figure.32 Book Flight Saudia

○ Data Variables

Arrays:

Basic/main data structures

- string usernameA[RECORDS];
- string passwordsA[RECORDS];
- string rolesA[RECORDS];
- int entry_count=0
- int flight_count=0
- int user_count=0
- int total_count=0

Passenger data structures

- string pName[RECORDS];
- string passport[RECORDS];
- string cnic[RECORDS];
- string mail[RECORDS];
- string gender[RECORDS];
- string pNum[RECORDS];
- string pClass[RECORDS];
- string pDepartCity[RECORDS];
- string pArrCity[RECORDS];
- string pTrip[RECORDS];
- string pDepartDate[RECORDS];
- int optionF[RECORDS]; // Flight Option
- int adult[RECORDS];
- int child[RECORDS];
- int infant[RECORDS];
- int totalSeats[RECORDS];
- int total[RECORDS]; // Expenditures
- string seating[RECORDS];

Admin data structures

- string aDepartDate[RECORDS];
- string aDepartCity[RECORDS];
- string aArrCity[RECORDS];
- string aTrip[RECORDS];
- string aDepartTime[RECORDS];

Functions:

○ Function Prototypes

1. `void header();`
2. `int loginMenu();`
3. `string signinMenu(string passwordsA[] ,string pass, string usernameA[], string &username, string rolesA[],const int &RECORDS);`
4. `void addUser(string &username, string password, string role,string passwordsA[], string usernameA[], string rolesA[],int &entry_count,int &total_count ,const int &RECORDS);`
5. `bool signup(string user,string pass,string role,string passwordsA[], string usernameA[], string rolesA[],int &entry_count,int &total_count ,const int &RECORDS);`
6. `void clearScreen();`
7. `int UserMenu();`
8. `void Register(string pName[], string passport[], string cnic[], string mail[], string gender[], string pNum[],int &entry_count ,const int &RECORDS);`
9. `void pDataIntoArray(string pname, string pcnic, string ppassport, string ppNum, string pmail, string pgender, string pName[], string passport[], string cnic[], string mail[], string gender[],string pNum[],int &entry_count);`
10. `void flightP(string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string pDepartDate[],int adult[],int child[],int infant[],int totalSeats[],int &entry_count ,const int &RECORDS);`
11. `void pFlightDataIntoArray(string departCity, string arrCity, string trip, string departDate, string pClassF, int adultF, int childF, int infantF, int totalSeatsF,string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string pDepartDate[],int adult[],int child[],int infant[],int totalSeats[],int &entry_count);`
12. `int bookP(string pDepartCity[],string pArrCity[],string pDepartDate[],string aDepartDate[],string aDepartCity[],string`

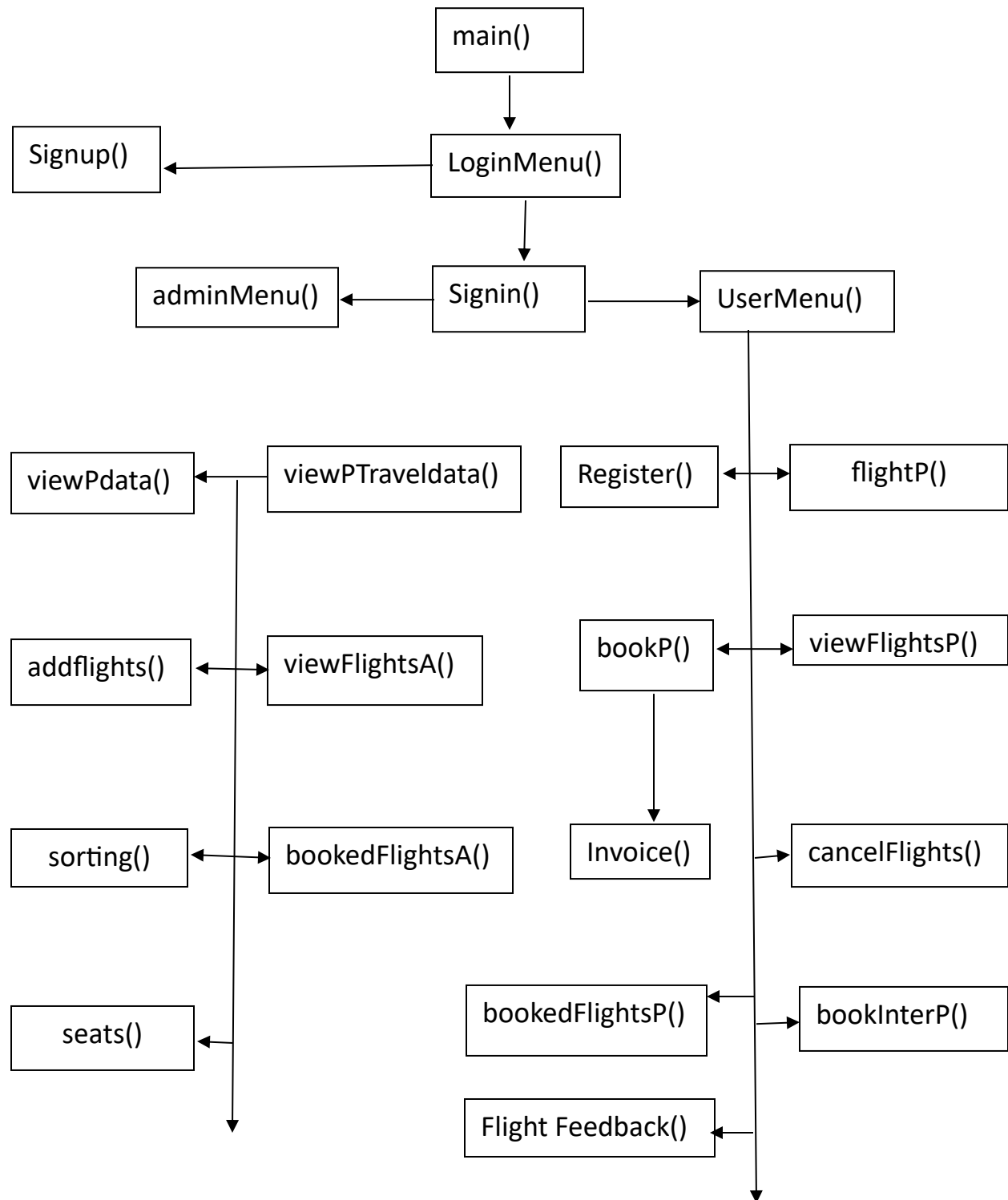
```
    aArrCity[],string aDepartTime[],int optionF[],int &entry_count ,int
    &flight_count );
13.void invoice(string pClass[],string pDepartCity[],string
    pArrCity[],string pTrip[],string pDepartDate[],int adult[],int
    child[],int infant[],int total[],int &entry_count );
14.void bookedFlightsP(string usernameA[], string &username, string
    rolesA[],string pDepartCity[],string pArrCity[],string
    pDepartDate[],int optionF[],int total[],const int &RECORDS,int&
    entry_count);
15.void cancelFlights(string usernameA[], string &username, string
    rolesA[],string pDepartCity[],string pArrCity[],string
    pDepartDate[],int optionF[],int total[],int &entry_count ,const int
    &RECORDS);
16.void viewFlightsP(string aDepartDate[],string aDepartCity[],string
    aArrCity[],string aTrip[],string aDepartTime[],int &flight_count );
17.int FlightFeedback( string flightNumber, string Name,int
    feedbackRating, string comments );
18.int bookInter();
19.int adminMenu();
20.void viewPData(string pName[], string passport[], string cnic[],
    string mail[], string gender[],string pNum[],string seating[],int
    total[],int &entry_count );
21.void viewPTravelData(string pClass[],string pDepartCity[],string
    pArrCity[],string pTrip[],string pDepartDate[],int adult[],int
    child[],int infant[],int &entry_count );
22.void addFlights(string aDepartDate[],string aDepartCity[],string
    aArrCity[],string aTrip[],string aDepartTime[],int &flight_count );
23.void viewFlightsA(string aDepartDate[],string aDepartCity[],string
    aArrCity[],string aTrip[],string aDepartTime[],int &flight_count );
24.void seats(string pClass[],string pDepartCity[],string
    pArrCity[],string pTrip[],string pDepartDate[],int adult[],int
    child[],int infant[],string seating[],int &entry_count );
25.void bookedFlightsA(string usernameA[], string &username, string
    rolesA[],string pDepartCity[],string pArrCity[],string
    pDepartDate[],int optionF[],int total[],const int &RECORDS);
26.void sorting(string pName[], string passport[], string cnic[],
    string mail[], string gender[],string pNum[],string seating[],int
    total[],int &entry_count );
27.void Astore(string aDepartCity[],string aArrCity[],string
    aTrip[],string aDepartDate[],string aDepartTime[],string pName[],
    string passport[], string cnic[], string mail[], string
    gender[],string pNum[],string pClass[],string pDepartCity[],string
```

```
pArrCity[],string pTrip[],string pDepartDate[],int adult[],int  
child[],int infant[]);  
28.void credentials( string &pass, string &username, string &role);  
29.int convert(string);  
30.string comma(string word, int field);
```

○ **Function Working Flow**

- ✓ I want to explain first. First admin will login and add flights to the system. Then he can view flight and then he can go to user menu to book flight. User will enter details and can book flight if available.
- ✓ Then he can book and take invoice and see booked flights.
- ✓ Now admin can see user information and travel details and see sorted passengers.

- FlowChart



- **Complete Code of the Business Application**

```
#include <iostream>
#include <conio.h>
#include <math.h>
#include <fstream>
#include <iomanip>
#include <cstdlib>
using namespace std;
void header();
void Astore(string aDepartCity[],string aArrCity[],string aTrip[],string
aDepartDate[],string aDepartTime[],string pName[], string passport[], string cnic[], string
mail[], string gender[],string pNum[],string pClass[],string pDepartCity[],string
pArrCity[],string pTrip[],string pDepartDate[],int adult[],int child[],int infant[]);
void credentials( string &pass, string &username, string &role);
int convert(string);
string comma(string word, int field);

//header
int loginMenu();
//loginmenu
string signinMenu(string passwordsA[] ,string pass, string usernameA[], string
&username, string rolesA[],const int &RECORDS);
//signin
void addUser(string &username, string password, string role,string passwordsA[], string
usernameA[], string rolesA[],int &entry_count,int &total_count ,const int &RECORDS);
//addrole
bool signup( string user,string pass,string role,string passwordsA[], string usernameA[],
string rolesA[],int &entry_count,int &total_count ,const int &RECORDS);
//signup
void clearScreen();
// Passenger Functions Prototypes
int UserMenu();
//1
void Register(string pName[], string passport[], string cnic[], string mail[], string
gender[], string pNum[],int &entry_count ,const int &RECORDS);
//1
void pDataIntoArray(string pname, string pcnic, string ppassport, string ppNum, string
pmail, string pgender, string pName[], string passport[], string cnic[], string mail[], string
gender[],string pNum[],int &entry_count );
```

```
//2
void flightP(string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string
pDepartDate[],int adult[],int child[],int infant[],int totalSeats[],int &entry_count ,const int
&RECORDS);
//2
void pFlightDataIntoArray(string departCity, string arrCity, string trip, string departDate,
string pClassF, int adultF, int childF, int infantF, int totalSeatsF,string pClass[],string
pDepartCity[],string pArrCity[],string pTrip[],string pDepartDate[],int adult[],int
child[],int infant[],int totalSeats[],int &entry_count );
//3
int bookP(string pDepartCity[],string pArrCity[],string pDepartDate[],string
aDepartDate[],string aDepartCity[],string aArrCity[],string aDepartTime[],int
optionF[],int &entry_count ,int &flight_count );
//3
void invoice(string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string
pDepartDate[],int adult[],int child[],int infant[],int total[],int &entry_count );
//4
void bookedFlightsP(string usernameA[], string &username, string rolesA[],string
pDepartCity[],string pArrCity[],string pDepartDate[],int optionF[],int total[],const int
&RECORDS);
//5
void cancelFlights(string usernameA[], string &username, string rolesA[],string
pDepartCity[],string pArrCity[],string pDepartDate[],int optionF[],int total[],int
&entry_count ,const int &RECORDS);
//6
void viewFlightsP(string aDepartDate[],string aDepartCity[],string aArrCity[],string
aTrip[],string aDepartTime[],int &flight_count );
//7
int FlightFeedback( string flightNumber, string Name,int feedbackRating, string
comments );
//8
int bookInter();
// Admin Functions Prototypes
int adminMenu();
//1
void viewPData(string pName[], string passport[], string cnic[], string mail[], string
gender[],string pNum[],string seating[],int total[],int &entry_count );
//2
void viewPTravelData(string pClass[],string pDepartCity[],string pArrCity[],string
pTrip[],string pDepartDate[],int adult[],int child[],int infant[],int &entry_count );
//3
```

```
void addFlights(string aDepartDate[],string aDepartCity[],string aArrCity[],string
aTrip[],string aDepartTime[],int &flight_count );
//4
void viewFlightsA(string aDepartDate[],string aDepartCity[],string aArrCity[],string
aTrip[],string aDepartTime[],int &flight_count );
//5
void seats(string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string
pDepartDate[],int adult[],int child[],int infant[],string seating[],int &entry_count );
//6
void bookedFlightsA(string usernameA[], string &username, string rolesA[],string
pDepartCity[],string pArrCity[],string pDepartDate[],int optionF[],int total[],const int
&RECORDS);
//7
void sorting(string pName[], string passport[], string cnic[], string mail[], string
gender[],string pNum[],string seating[],int total[],int &entry_count );

main()
{

    // Count Variables
    const int RECORDS = 200;
    int entry_count = -1, flight_count = 1, user_count = 0, total_count = 0;
    // BASIC/MAIN DATA STRUCTURES
    string usernameA[RECORDS];
    string passwordsA[RECORDS];
    string rolesA[RECORDS];
    string username;
    // USER DATA STRUCTURES
    string pName[RECORDS];
    string passport[RECORDS];
    string cnic[RECORDS];
    string mail[RECORDS];
    string gender[RECORDS];
    string pNum[RECORDS];
    string pClass[RECORDS];
    string pDepartCity[RECORDS];
    string pArrCity[RECORDS];
    string pTrip[RECORDS];
    string pDepartDate[RECORDS];
    int optionF[RECORDS]; // Flight Option
    int adult[RECORDS];
```



```
int child[RECORDS];
int infant[RECORDS];
int total[RECORDS]; // Expenditures
string seating[RECORDS];
// ADMIN DATA STRUCTURES
string aDepartDate[RECORDS];
string aDepartCity[RECORDS];
string aArrCity[RECORDS];
string aTrip[RECORDS];
string aDepartTime[RECORDS];
int totalSeats[RECORDS];

//
string flightNumber="SV-734";
string Name="Huzaifa";
string comments;
int feedbackRating;
string user, pass, role;

string who = " ";
// Options to select, Variable.
int option = 0,
    optionP = 0,
    optionA = 0;

while (true)
{
    system("color 02");
    header();
    option = loginMenu();
    if (option == 1)
    {
        header();
        who = signinMenu(passwordsA, pass, usernameA, username, rolesA
,RECORDS);

        // Passenger
        if (who == "user" )
        {
            system("cls");
```

```
optionP = 0;
while (optionP < 9)
{
    optionP = UserMenu();
    // REGISTRATION
    if (optionP == 1)
    {
        system("cls");
        Register(pName, passport, cnic,
mail,gender,pNum,entry_count,RECORDS);
        clearScreen();
    }
    // Enter Flight Details
    else if (optionP == 2)
    {
        system("cls");

flightP(pClass,pDepartCity,pArrCity,pTrip,pDepartDate,adult,child,infant,totalSeats,entry
_count,RECORDS);
        clearScreen();
    }
    // Book from available Flights
    else if (optionP == 3)
    {
        system("cls");
        optionF[entry_count] = bookP(
pDepartCity,pArrCity,pDepartDate,aDepartDate,aDepartCity,aArrCity,aDepartTime,optio
nF,entry_count,flight_count );
        if (optionF[entry_count] == 1)
        {

invoice(pClass,pDepartCity,pArrCity,pTrip,pDepartDate,adult,child,infant,total,entry_co
unt);
        }
        clearScreen();
    }
    // View booked flights
    else if (optionP == 4)
    {
        system("cls");
```

```
        bookedFlightsP(usernameA, username, rolesA,
pDepartCity,pArrCity,pDepartDate,optionF,total,RECORDS);
        clearScreen();
    }
    // Cancel booked flights
    else if (optionP == 5)
    {
        system("cls");
        cancelFlights(usernameA, username,
rolesA,pDepartCity,pArrCity,pDepartDate,optionF,total,entry_count,RECORDS);
        clearScreen();
    }
    //view flights
    else if (optionP == 6)
    {
        system("cls");
        viewFlightsP(aDepartDate,aDepartCity,aArrCity,
aTrip,aDepartTime,flight_count );
        clearScreen();
    }
    //feedback
    else if (optionP == 7)
    {
        system("cls");
        int ans = FlightFeedback(
flightNumber,Name,feedbackRating,comments);
        // Display feedback
        cout << "\nThank you for providing feedback!\n";
        cout << "Flight: " << flightNumber << "\nPassenger: " << Name << "\n";
        cout << "Your rating: " << ans;
        cout << "Your Feedback is added to our system.." << endl;
        cout << endl;

        clearScreen();
    }
    //book to international flight
    else if( optionP == 8)
    {
        system("cls");
        bookInter();
        clearScreen();
    }
}
```

```
    }
  }
}

// ADMIN
else if (who == "admin")
{
    optionA = 0;
    while (optionA < 9)
    {
        system("cls");
        optionA = adminMenu();
        // View Passengers Data
        if (optionA == 1)
        {
            system("cls");
            viewPData( pName, passport, cnic, mail, gender,
pNum,seating,total,entry_count);
            clearScreen();
        }
        // View Passengers travel Data
        else if (optionA == 2)
        {
            system("cls");

viewPTravelData(pClass,pDepartCity,pArrCity,pTrip,pDepartDate,adult,child,infant,entr
y_count);
            clearScreen();
        }
        // Add new flights
        else if (optionA == 3)
        {
            system("cls");
            addFlights(aDepartDate,aDepartCity,aArrCity,
aTrip,aDepartTime,flight_count );
            clearScreen();
        }
        //view flights that admin has added
        else if (optionA == 4)
        {
            system("cls");
```

```
        viewFlightsA(aDepartDate,aDepartCity,aArrCity,
aTrip,aDepartTime,flight_count );
        clearScreen();
    }

    // Allot Seats
    else if (optionA == 5)
    {
        system("cls");

seats(pClass,pDepartCity,pArrCity,pTrip,pDepartDate,adult,child,infant,seating,entry_co
unt);
        clearScreen();
    }
    //view booked flights by passenger
    else if (optionA == 6)
    {
        system("cls");
        bookedFlightsA(usernameA, username,
rolesA,pDepartCity,pArrCity,pDepartDate,optionF,total,RECORDS);
        clearScreen();
    }

    // Ordered Passengers
    else if (optionA == 7)
    {
        system("cls");
        sorting( pName, passport, cnic, mail,
gender,pNum,seating,total,entry_count);
        clearScreen();
    }
    else if (optionA == 8)
    {
        system("cls");
        Astore(aDepartCity,aArrCity,aTrip,aDepartDate,aDepartTime,
pName,passport,cnic,mail,gender,pNum, pClass, pDepartCity, pArrCity,
pTrip,pDepartDate,adult,child,infant);
        clearScreen();
    }
}
```

```
    }
    else if (who == "EXIT")
    {
        cout << "You entered wrong credentials.." << endl;
        clearScreen();
    }
}

if (option == 2)
{
    header();
    cout << "Main Menu > Sign Up " << endl;
    cout << "-----" << endl;
    signup(user,pass,role,passwordsA, usernameA, rolesA,entry_count,total_count
,RECORDS);
    clearScreen();
}
if(option == 3)
{
    header();
    cout << "Main Menu > User Details" << endl;
    cout << "-----" << endl;
    credentials(pass,username,role);
    clearScreen();
}
if(option==4)
{
    return 0;
}

}

}

// END OF MAIN FUNCTION

void header()
{

    system("cls");
```

[illegible]

```
    cout<<"
    "<<endl;
    cout<<R"(------)
    -----)"<<endl;

}
void clearScreen()
{
    cout << "Press any key to continue " << endl;
    getch();
    system("cls");
}

// Login MENU
int loginMenu()
{
    int option;
    cout << " -----" << endl;
    cout << "|                               | " << endl;
    cout << "|----- LOGIN MENU -----| " <<
endl;
    cout << "|                               | " << endl;
    cout << "|                               | " << endl;
    cout << "|----- " << endl;
    cout <<endl;
    cout << "Login Menu " << endl;
    cout << "-----" << endl;
    cout << "Select one of the following options..." << endl;

    cout << "1- Sign In" << endl;
    cout << "2- Sign Up" << endl;
    cout << "3- User Details" << endl;
    cout << "4- Exit" << endl;
    cout << "Your Option..";
    while(!(cin>>option))
    {
        cout << "Your Option..";
        cin.clear();
        cin.ignore();
    }
    return option;
}
```



```
}
string signinMenu(string passwordsA[] ,string pass, string usernameA[], string
&username, string rolesA[],const int &RECORDS)
{

    cout << "Main Menu > Login" << endl;
    cout << "-----" << endl;
    cout << "Enter username :- ";
    getline(cin >>ws,username);
    cout << "Enter password :- ";
    getline(cin>>ws,pass);

    for (int i = 0; i < RECORDS; i++)
    {
        if (usernameA[i] == username && passwordsA[i] == pass)
        {
            return rolesA[i];
        }
    }

    return "EXIT";
}
bool signup(string user,string pass, string role,string passwordsA[], string usernameA[],
string rolesA[],int &entry_count,int &total_count,const int &RECORDS )
{
    bool flag = true;
    int c = 0, x;
    char ch;

    cout << "Enter the username: " << endl;
    getline(cin>>ws,user);
    while (flag)
    {
        cout << "Enter the password: " << endl;
        getline(cin>>ws,pass);
        c = 0;
        for (int i = 0; i < pass.length(); i++)
        {
            ch = pass[i];
```

```
        x = ch;
        if (pass[i] >= 48 && pass[i] <= 57)
        {
            c++;
        }
    }
    if (c == 0 || pass.length() < 2)
    {
        flag = true;
        cout << "Password Should Contain a Number and should be more than 1
character" << endl;
    }
    else
    {
        flag = false;
    }
}

cout << "Enter the role: " << endl;
getline(cin >>ws,role);
addUser(user, pass, role ,passwordsA, usernameA, rolesA,entry_count,total_count
,RECORDS);
}
void addUser(string &username, string password, string role,string passwordsA[], string
usernameA[], string rolesA[],int &entry_count,int &total_count,const int &RECORDS)
{

    if (total_count < RECORDS)
    {
        usernameA[total_count] = username;
        passwordsA[total_count] = password;
        rolesA[total_count] = role;
        if (role == "user")
        {
            entry_count++;
        }

        total_count++;

        cout
```

```
<< endl
<< "Registered Successfully . You could sign in now" << endl
<< endl;
}
else
{
    clearScreen();
}
ofstream file("User.txt", ios::app);
file << username << "," << password << "," << role << endl;
}
// MENU OF PASSENGER
int UserMenu()
{

    system ("color 02");
    cout << " -----" << endl;
    cout << "|                                     | " << endl;
    cout << "|----- USER MENU -----| " <<
endl;
    cout << "|                                     | " << endl;
    cout << "|                                     | " << endl;
    cout << "|----- " << endl;
    cout << endl;

    int optionP;
    cout << "Login Menu > Login > Passenger" << endl;
    cout << "-----" << endl;
    cout << "Select one of the following options..." << endl;
    cout << "1- Registration" << endl;
    cout << "2- Enter Flight Details" << endl;
    cout << "3- Book from available Flights" << endl;
    cout << "4- See booked Flights " << endl;
    cout << "5- Cancel A flight " << endl;
    cout << "6- View Flights " << endl;
    cout << "7- Provide feedback " << endl;
    cout << "8- Booking to Differnt Country " << endl;
    cout << "9- Exit " << endl;
    cout << "Your Option..";
    while(!(cin>>optionP))
    {
```

```
        cout << "Your Option..";
        cin.clear();
        cin.ignore();
    }
    return optionP;
}
// 1 REGISTRATION OF PASSENGER
void Register(string pName[], string passport[], string cnic[], string mail[], string
gender[],string pNum[],int &entry_count ,const int &RECORDS)
{
    string ppname,ppcnic,ppNum,ppmail,ppgender, ppassport;
    if (entry_count < RECORDS)
    {
        cout << "Login Menu > Login > Passenger > Registration" << endl;
        cout << "-----" << endl;

        cout << "Enter your name :- " << endl;
        getline(cin>>ws,ppname);
        cout << "Enter your CNIC :- " << endl;
        getline(cin>>ws,ppcnic);
        cout << "Enter your passport no. :- " << endl;
        getline(cin>>ws,pppassport);
        cout << "Enter your Phone Number :- " << endl;
        getline(cin>>ws,ppNum);
        cout << "Enter your email :- " << endl;
        getline(cin>>ws,ppmail);
        cout << "Enter your Gender :- " << endl;
        getline(cin>>ws,ppgender);
        pDataIntoArray(ppname, ppcnic, ppassport, ppNum, ppmail, ppgender, pName,
passport, cnic, mail, gender, pNum,entry_count);
    }
    else
    {
        cout << "Your Safety is our Priority";
    }
    ofstream file("PassengerData.txt",ios::app);
    file
    <<ppname<<","<<ppcnic<<","<<pppassport<<","<<ppNum<<","<<ppmail<<","<<ppgen
der<<endl;
    file.close();
}
```

```
void pDataIntoArray(string pname, string pcnic, string ppassport, string ppNum, string
pmail, string pgender, string pName[], string passport[], string cnic[], string mail[], string
gender[],string pNum[],int &entry_count )
{

    pName[entry_count] = pname;
    cnic[entry_count] = pcnic;
    passport[entry_count] = ppassport;
    pNum[entry_count] = ppNum;
    mail[entry_count] = pmail;
    gender[entry_count] = pgender;
}
// 2 FLIGHT DETAILS OF PASSENGER
void flightP(string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string
pDepartDate[],int adult[],int child[],int infant[],int totalSeats[],int &entry_count ,const int
&RECORDS)
{
    string ppdepartCity, pparrCity, pptrip,ppdepartDate,ppClass;
    int ppadult, ppchild,ppinfant, pptotalSeats;
    if (entry_count < RECORDS)
    {
        cout << "Login Menu > Login > Passenger > Enter Flight Details" << endl;
        cout << "-----" << endl;
        cout << "Departure from :- " << endl;
        getline(cin>>ws,ppdepartCity);
        cout << "Arrival To :- " << endl;
        getline(cin>>ws,pparrCity);
        cout << "Trip Type :- " << endl;
        getline(cin>>ws,pptrip);
        cout << "Depart Date :-" << endl;
        getline(cin>>ws,ppdepartDate);
        cout << "-----Passengers-----" << endl;
        cout << "No of Passengers (Adult) :- " << endl;
        while(!(cin>>ppadult))
        {
            cout << "No of Passengers (Adult) :- "<<endl;
            cin.clear();
            cin.ignore();
        }
        cout << "No of Passengers (Child) :- " << endl;
```

```
while(!(cin>>ppchild))
{
    cout << "No of Passengers (Child) :- " << endl;
    cin.clear();
    cin.ignore();
}
cout << "No of Passengers (Infant/baby) :- " << endl;
while(!(cin>>ppinfant))
{
    cout << "No of Passengers (Infant) :- " << endl;
    cin.clear();
    cin.ignore();
}
cout << "-----Class-----" << endl;
cout << "Enter Class :- " << endl;
getline(cin>>ws, ppClass);
pptotalSeats = ppadult + ppchild;
pFlightDataIntoArray(ppdepartCity, pparrCity, pptrip, ppdepartDate, ppClass,
ppadult, ppchild, ppinfant, pptotalSeats, pClass, pDepartCity, pArrCity,
pTrip, pDepartDate, adult, child, infant, totalSeats, entry_count);
}
else
{
    cout << "Your safety is our priority";
}

ofstream file("PassengerTravel.txt", ios::app);
file
<<ppdepartCity<<","<<pparrCity<<","<<pptrip<<","<<ppdepartDate<<","<<ppadult<<
,<<ppchild<<","<<ppinfant<<","<<ppClass<<endl;
file.close();
}

void pFlightDataIntoArray(string departCity, string arrCity, string trip, string departDate,
string pClassF, int adultF, int childF, int infantF, int totalSeatsF, string pClass[], string
pDepartCity[], string pArrCity[], string pTrip[], string pDepartDate[], int adult[], int
child[], int infant[], int totalSeats[], int &entry_count )
{

    pDepartCity[entry_count] = departCity;
    pArrCity[entry_count] = arrCity;
    pTrip[entry_count] = trip;
```

```
pDepartDate[entry_count] = departDate;
pClass[entry_count] = pClassF;
adult[entry_count] = adultF;
child[entry_count] = childF;
infant[entry_count] = infantF;
totalSeats[entry_count] = totalSeatsF;
}
// 3 Book from available Flights
int bookP(string pDepartCity[],string pArrCity[],string pDepartDate[],string
aDepartDate[],string aDepartCity[],string aArrCity[],string aDepartTime[],int
optionF[],int &entry_count,int &flight_count )
{

    cout << "Login Menu > Login > Passenger > Book from available Flights" <<
endl;
    cout << "-----" << endl;
    for (int i = 0; i < flight_count; i++)
    {
        if (pDepartCity[entry_count] == aDepartCity[i] && pArrCity[entry_count] ==
aArrCity[i] && pDepartDate[entry_count] == aDepartDate[i])
        {
            cout << "1- From \t " << pDepartCity[entry_count] << "\t To \t " <<
pArrCity[entry_count] << " \t at \t " << aDepartTime[i] << "\t \t " <<
pDepartDate[entry_count] << endl;
            while(!(cin >> optionF[entry_count]))
            {
                cout << "1- From \t " << pDepartCity[entry_count] << "\t To \t " <<
pArrCity[entry_count] << " \t at \t " << aDepartTime[i] << "\t \t " <<
pDepartDate[entry_count] << endl;
                cin.clear();
                cin.ignore();
            }
            return optionF[entry_count];
            flight_count++;
        }
        else
        {
            optionF[entry_count] = 0;
        }
    }
}
```

```
    return optionF[entry_count];
}
// 4 ticket
void invoice(string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string
pDepartDate[],int adult[],int child[],int infant[],int total[],int &entry_count )
{

    cout << endl
        << "Departure "
        << "\t"
        << "Arrival"
        << "\t"
        << "Trip"
        << "\t"
        << "Date"
        << "\t"
        << "Adults"
        << "\t"
        << "Child"
        << "\t"
        << "Infants"
        << "\t"
        << "Class " << endl;
    cout << pDepartCity[entry_count] << "\t\t" << pArrCity[entry_count] << "\t" <<
pTrip[entry_count] << "\t" << pDepartDate[entry_count] << "\t" << adult[entry_count]
<< "\t" << child[entry_count] << "\t" << infant[entry_count] << "\t" <<
pClass[entry_count] << endl;

    cout << endl
        << "-----Expenditures-----" << endl;

    cout << "No of Passengers (Adult)(400$) :- " << adult[entry_count] << endl;
    cout << "No of Passengers (Child) (250$):- " << child[entry_count] << endl;
    cout << "No of Passengers (Infant/baby)(0$) :- " << infant[entry_count] << endl
        << endl;
    total[entry_count] = (adult[entry_count] * 400) + (child[entry_count] * 250);
    if (pClass[entry_count] == "Business" || pClass[entry_count] == "business")
    {
        total[entry_count] = total[entry_count] + 200;
    }
    cout << "Your Total Expenditure :- " << total[entry_count] << "$" << endl;
```



```
}
// 5 See booked Flights
void bookedFlightsP(string usernameA[], string &username, string rolesA[],string
pDepartCity[],string pArrCity[],string pDepartDate[],int optionF[],int total[],const int
&RECORDS)
{

    int i = 0;
    cout << "Login Menu > Login > Passenger > See booked Flights" << endl;
    cout << "-----" << endl;
    cout << "Your Flights..." << endl;
    for (int j = 0; j < RECORDS; j++)
    {
        if (rolesA[j] == "admin")
        {
            i--;
        }
        else if (usernameA[j] == username)
        {
            i = j + i;
            // cout << j << endl
            // << i << endl;
            if (optionF[i] == 1)
            {
                cout << "1- From \t " << pDepartCity[i] << "\t To \t " << pArrCity[i] << " \t at
\t 12:00 \t \t " << pDepartDate[i] << endl;
                cout << "Your Total Expenditure :- " << total[i] << endl;
            }
            else if (optionF[i] == 0)
            {
                cout << "No booked flights yet!!" << endl;
            }
        }
    }
}

// 6 Cancel Flights
void cancelFlights(string usernameA[], string &username, string rolesA[],string
pDepartCity[],string pArrCity[],string pDepartDate[],int optionF[],int total[],int
&entry_count ,const int &RECORDS)
{
```

```
int i = 0;
int cancel;
cout << "Login Menu > Login > Passenger > Cancel A flight" << endl;
cout << "-----" << endl;
cout << "Your Flights..." << endl
    << endl;
for (int j = 0; j < RECORDS; j++)
{
    if (rolesA[j] == "admin")
    {
        i--;
    }
    else if (usernameA[j] == username)
    {
        i = j + i;
        if (optionF[i] == 1)
        {
            cout << "1- From \t " << pDepartCity[i] << "\t To \t " << pArrCity[i] << "\t at
\t 12:00 \t \t " << pDepartDate[i] << endl;
            cout << "Your Total Expenditure :- " << total[i] << endl;
            cout << "Your Option..";
            while(!(cin>>cancel))
            {
                cout << "Your Option..";
                cin.clear();
                cin.ignore();
            }
            if (cancel == 1)
            {
                cout << "Flight has been cancelled!!"<<endl;
                optionF[entry_count] = 0;
            }
        }

        else if (optionF[i] == 0)
        {
            cout << "No booked flights yet!!" << endl
                << endl;
        }
    }
}
```

```
}
// 7 view flights
void viewFlightsP(string aDepartDate[],string aDepartCity[],string aArrCity[],string
aTrip[],string aDepartTime[],int &flight_count )
{
    cout << flight_count << endl;

    cout << "Login Menu > Login > Passenger > View Flights" << endl;
    cout << "-----" << endl;
    cout << "Departure"
        << "\t"
        << "Arrival"
        << "\t"
        << "Trip"
        << "\t"
        << "DepartDate"
        << "\t"
        << "DepartTime"
        << endl;

    string record;
    ifstream file("admin1.txt");
    while(getline(file,record))
    {
        cout<<comma(record,0)<<"\t"<<comma(record,1)<<
"\t"<<comma(record,2)<<"\t"<<comma(record,3)<<"\t"<<comma(record,4)<<endl;
    }
    file.close();

}
// 8 feedback
int FlightFeedback( string flightNumber, string Name,int feedbackRating, string
comments )
{

    Name="Huzaifa";
    flightNumber="SV-734";
    // Assume a rating scale from 1 to 5
    cout << "Login Menu > Login > Passenger > Feedback" << endl;
    cout << "-----" << endl;
    cout<<endl;
```

```
    cout << "Please provide feedback for flight " << flightNumber << " for passenger " <<
    Name << ":" << endl;
    cout << endl;
    cout << "Comments: " << endl;
    getline(cin >> ws, comments);

    // Get rating
    do {
        cout << "Rating (1 to 5): ";
        while (!(cin >> feedbackRating))
        {
            cout << "Rating (1 to 5): ";
            cin.clear();
            cin.ignore();
        }

        // Validate rating
        if (feedbackRating < 1 || feedbackRating > 5) {
            cout << "Invalid rating. Please enter a number between 1 and 5." << endl;
        }
        else {
            clearScreen();
        }
    }
    while (feedbackRating < 1 || feedbackRating > 5);
    return feedbackRating;
}

// 9 Book to International countries
int bookInter()
{
    int back;
    int choice;
    int choice1;
    float charges;
    string flightN[] =
    {"USA", "UK", "Germany", "Australia", "Pakistan", "Qatar", "Turkey", "Dubai", "Korea", "Sa
    udi Airlines"};
    cout << "Login Menu > Login > Passenger > International flights" << endl;
    cout << "-----" << endl;

    for(int a=0; a<10; a++)
```

```
{
    cout<<(a+1)<<".flight to "<<flightN[a]<<endl;
}
cout<<"\nWelcome to the Airlines! "<<endl;
cout<<endl;
cout<<endl;
cout<<"_____Booking_____ "<<endl;
cout<<endl;
cout<<"Enter the Number of country you want to book the flight: ";
while(!(cin>>choice))
{
    cout<<"Enter the Number of country you want to book the flight: ";
    cin.clear();
    cin.ignore();
}

system("cls");
switch(choice)
{
    case 1:
    {
        cout<<"_____Welcome to USA Airlines_____ \n"<<endl;
        cout<<endl;
        cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
        cout<<"Following are the flights\n"<<endl;
        cout<<"1. USA - 498"<<endl;
        cout<<"\t01-01-2024 8:00AM 10hrs 1000$"<<endl;

        cout<<"2. USA - 658"<<endl;
        cout<<"\t09-01-2024 12:00AM 11hrs 1100$"<<endl;

        cout<<"3. USA - 704"<<endl;
        cout<<"\t11-01-2024 7:00PM 10hrs 1250$"<<endl;

        cout<<"Select the flight you want to Book: "<<endl;
        while(!(cin>>choice1))
        {
            cout<<"Select the flight you want to Book: "<<endl;
            cin.clear();
            cin.ignore();
        }
    }
}
```

```
if(choice1==1)
{
    charges=1000;
    cout<<"\nYou have Sucessfully Booked the flight USA - 498"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;

}
else if(choice1==2)
{
    charges=1100;
    cout<<"\nYou have Sucessfully Booked the flight USA - 658"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;

}
else if(choice1==3)
{
    charges=1250;
    cout<<"\nYou have Sucessfully Booked the flight USA - 704"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;

}
else
{
    cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
    clearScreen();
    UserMenu();

}

cout<<"Press 1 to go back to Menu.."<<endl;
cin>>back;
system("cls");
if(back==1)
{

    UserMenu();

}
```

```
        break;
    }

    else
    {

        UserMenu();
        break;
    }
}
break;

case 2:
{
    cout<<"_____Welcome to British
Airlines_____\\n"<<endl;
    cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
    cout<<"Following are the flights\\n"<<endl;
    cout<<"1. UK - 233"<<endl;
    cout<<"\\t09-02-2024 8:00PM 8hrs 800 euro"<<endl;

    cout<<"2. UK - 467"<<endl;
    cout<<"\\t08-02-2024 12:00AM 11hrs 950 euro"<<endl;

    cout<<"3. UK - 706"<<endl;
    cout<<"\\t11-01-2024 7:00PM 10hrs 750 euro"<<endl;

    cout<<"Select the flight you want to Book: ";
    while(!(cin>>choice1))
    {
        cout<<"Select the flight you want to Book: "<<endl;
        cin.clear();
        cin.ignore();
    }

    if(choice1==1)
    {
        charges=800;
        cout<<"\\nYou have Sucessfully Booked the flight UK - 233 "<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;
    }
}
```

```
    }
    else if(choice1==2)
    {
        charges=950;
        cout<<"\nYou have Sucessfully Booked the flight UK- 467"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;

    }
    else if(choice1==3)
    {
        charges=750;
        cout<<"\nYou have Sucessfully Booked the flight UK - 706"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;

    }
    else
    {
        cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
        clearScreen();
        UserMenu();

    }

    cout<<"Press 1 to go back to Menu.."<<endl;
    cin>>back;
    system("cls");
    if(back==1)
    {

        UserMenu();
        break;
    }

    else
    {

        clearScreen();
        UserMenu();
    }
}
```



```
        break;
    }

}

case 3:

    {   cout<<"_____Welcome to German
Airlines_____\\n"<<endl;
        cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
        cout<<"Following are the flights\\n"<<endl;
        cout<<"1. GRM - 498"<<endl;
        cout<<"\\t01-03-2024 08:00AM 10hrs 300 euro"<<endl;

        cout<<"2. GRM - 658"<<endl;
        cout<<"\\t09-04-2024 12:00AM 11hrs 500 euro"<<endl;

        cout<<"3. GRM - 704"<<endl;
        cout<<"\\t11-05-2024 07:00PM 10hrs 700 euro"<<endl;

        cout<<"Select the flight you want to Book: ";
        while(!(cin>>choice1))
        {
            cout<<"Select the flight you want to Book: "<<endl;
            cin.clear();
            cin.ignore();
        }

        if(choice1==1)
        {
            charges=300;
            cout<<"\\nYou have Sucessfully Booked the flight GRM - 498"<<endl;
            cout<<"You can go back to menu and take the ticket"<<endl;
            cout<<endl;
        }
        else if(choice1==2)
        {
            charges=500;
            cout<<"\\nYou have Sucessfully Booked the flight GRM - 658"<<endl;
            cout<<"You can go back to menu and take the ticket"<<endl;
```

```
        cout<<endl;

    }
    else if(choice1==3)
    {
        charges=700;
        cout<<"\nYou have Sucessfully Booked the flight GRM - 704"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;

    }
    else
    {
        cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
        clearScreen();
        UserMenu();

    }

    cout<<"Press 1 to go back to Menu.."<<endl;
    cin>>back;
    system("cls");
    if(back==1)
    {

        UserMenu();
        break;
    }

    else
    {

        clearScreen();
        UserMenu();
    }

}
case 4:
{
```

```
cout<<"_____Welcome to AUS Airlines_____"<<endl;
cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
cout<<"Following are the flights\n"<<endl;
cout<<"1. AUS - 499"<<endl;
cout<<"\t06-05-2024 07:00AM 10hrs 789$"<<endl;

cout<<"2. AUS - 652"<<endl;
cout<<"\t07-05-2024 11:00AM 11hrs 876$"<<endl;

cout<<"3. AUS - 728"<<endl;
cout<<"\t13-06-2024 05:00PM 10hrs 675$"<<endl;

cout<<"Select the flight you want to Book: ";
while(!(cin>>choice1))
{
    cout<<"Select the flight you want to Book: "<<endl;
    cin.clear();
    cin.ignore();
}

if(choice1==1)
{
    charges=789;
    cout<<"\nYou have Sucessfully Booked the flight AUS - 499"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==2)
{
    charges=876;
    cout<<"\nYou have Sucessfully Booked the flight AUS - 652"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==3)
{
    charges=675;
    cout<<"\nYou have Sucessfully Booked the flight AUS - 728"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
```

```
    }
    else
    {
        cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
        clearScreen();
        UserMenu();
    }

    cout<<"Press 1 to go back to Menu.."<<endl;
    cin>>back;
    system("cls");
    if(back==1)
    {

        UserMenu();
        break;
    }

    else
    {

        clearScreen();
        UserMenu();
    }
}
case 5:
{
    cout<<"_____ Welcome to PAK Airlines _____\n"<<endl;
    cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
    cout<<"Following are the flights\n"<<endl;
    cout<<"1. PIA - 445"<<endl;
    cout<<"\t04-07-2024 8:00AM 10hrs Rs.50400"<<endl;

    cout<<"2. PIA - 667"<<endl;
    cout<<"\t09-01-2024 12:00AM 11hrs Rs.45000"<<endl;
```

```
cout<<"3. PIA - 899"<<endl;
cout<<"\t11-03-2024 7:00PM 10hrs Rs.87000"<<endl;

cout<<"Select the flight you want to Book: ";
while(!(cin>>choice1))
{
    cout<<"Select the flight you want to Book: "<<endl;
    cin.clear();
    cin.ignore();
}

if(choice1==1)
{
    charges=50400;
    cout<<"\nYou have Sucessfully Booked the flight PIA - 445"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==2)
{
    charges=45000;
    cout<<"\nYou have Sucessfully Booked the flight PIA - 667"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==3)
{
    charges=87000;
    cout<<"\nYou have Sucessfully Booked the flight PIA - 899"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else
{
    cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
    clearScreen();
    UserMenu();
}
```

```
        cout<<"Press 1 to go back to Menu.."<<endl;
        cin>>back;
        system("cls");
        if(back==1)
        {

            UserMenu();
            break;
        }

        else
        {

            clearScreen();
            UserMenu();
        }

    }
    case 6:
    {
        cout<<"_____Welcome to Qatar
Airlines_____\\n"<<endl;
        cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
        cout<<"Following are the flights\\n"<<endl;
        cout<<"1. QTR - 447"<<endl;
        cout<<"\t04-05-2024 08:00AM 10hrs 2509 Riyal"<<endl;

        cout<<"2. QTR - 657"<<endl;
        cout<<"\t09-08-2024 12:00AM 11hrs 2550 Riyal"<<endl;

        cout<<"3. QTR - 789"<<endl;
        cout<<"\t27-03-2024 07:00PM 10hrs 2540 Riyal"<<endl;

        cout<<"Select the flight you want to Book: ";
        while(!(cin>>choice1))
        {
            cout<<"Select the flight you want to Book: "<<endl;
            cin.clear();
        }
    }
}
```

```
        cin.ignore();
    }

    if(choice1==1)
    {
        charges=2509;
        cout<<"\nYou have Sucessfully Booked the flight QTR - 447"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;
    }
    else if(choice1==2)
    {
        charges=2550;
        cout<<"\nYou have Sucessfully Booked the flight QTR - 657"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;
    }
    else if(choice1==3)
    {
        charges=2540;
        cout<<"\nYou have Sucessfully Booked the flight QTR - 789"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;
    }
    else
    {
        cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
        clearScreen();
        UserMenu();
    }

    cout<<"Press 1 to go back to Menu.."<<endl;
    cin>>back;
    system("cls");
    if(back==1)
    {
```

```
        UserMenu();
        break;
    }

    else
    {

        clearScreen();
        UserMenu();
    }
}
case 7:
{
    cout<<"_____Welcome to Turkish
Airlines_____\\n"<<endl;
    cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
    cout<<"Following are the flights\\n"<<endl;
    cout<<"1. TKR - 434"<<endl;
    cout<<"\\t17-05-2024 08:00AM 10hrs 2700 Liras"<<endl;

    cout<<"2. TKR - 783"<<endl;
    cout<<"\\t14-08-2024 12:00AM 11hrs 3000 Liras"<<endl;

    cout<<"3. TKR - 985"<<endl;
    cout<<"\\t03-03-2024 07:00PM 10hrs 4578 Liras"<<endl;

    cout<<"Select the flight you want to Book: ";
    while(!(cin>>choice1))
    {
        cout<<"Select the flight you want to Book: "<<endl;
        cin.clear();
        cin.ignore();
    }

    if(choice1==1)
    {
        charges=2700;
        cout<<"\\nYou have Sucessfully Booked the flight TKR - 434"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;
```



```
}
else if(choice1==2)
{
    charges=3000;
    cout<<"\nYou have Sucessfully Booked the flight TKR - 783"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;

}
else if(choice1==3)
{
    charges=4578;
    cout<<"\nYou have Sucessfully Booked the flight TKR - 985"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;

}
else
{
    cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
    clearScreen();
    UserMenu();

}

cout<<"Press 1 to go back to Menu.."<<endl;
cin>>back;
system("cls");
if(back==1)
{

    UserMenu();
    break;
}

else
{

    clearScreen();
```

```
        UserMenu();
    }

}

case 8:
{
    cout<<"_____Welcome to Dubai
Emirates_____\\n"<<endl;
    cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
    cout<<"Following are the flights\\n"<<endl;
    cout<<"1. DUB - 474"<<endl;
    cout<<"\\t17-04-2024 08:00AM 10hrs 2700 Riyals"<<endl;

    cout<<"2. DUB - 703"<<endl;
    cout<<"\\t14-05-2024 12:00AM 11hrs 3000 Riyals"<<endl;

    cout<<"3. DUB - 905"<<endl;
    cout<<"\\t03-02-2024 07:00PM 10hrs 4578 Riyals"<<endl;

    cout<<"Select the flight you want to Book: ";
    while(!(cin>>choice1))
    {
        cout<<"Select the flight you want to Book: "<<endl;
        cin.clear();
        cin.ignore();
    }

    if(choice1==1)
    {
        charges=2700;
        cout<<"\\nYou have Sucessfully Booked the flight DUB - 474"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;
    }
    else if(choice1==2)
    {
        charges=3000;
        cout<<"\\nYou have Sucessfully Booked the flight DUB - 703"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;
    }
}
```

```
    }
    else if(choice1==3)
    {
        charges=4578;
        cout<<"\nYou have Sucessfully Booked the flight DUB - 905"<<endl;
        cout<<"You can go back to menu and take the ticket"<<endl;
        cout<<endl;

    }
    else
    {
        cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
        clearScreen();
        UserMenu();
    }

    cout<<"Press 1 to go back to Menu.."<<endl;
    cin>>back;
    system("cls");
    if(back==1)
    {
        UserMenu();
        break;
    }

    else
    {
        clearScreen();
        UserMenu();
    }

}
case 9:
{
    cout<<"_____Welcome to Korean
Airlines_____ \n"<<endl;
```

```
cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
cout<<"Following are the flights\n"<<endl;
cout<<"1. KR - 774"<<endl;
cout<<"\t15-04-2024 08:00AM 10hrs 2708 won"<<endl;

cout<<"2. KR - 103"<<endl;
cout<<"\t15-05-2024 12:00AM 11hrs 3009 won"<<endl;

cout<<"3. KR - 105"<<endl;
cout<<"\t05-02-2024 07:00PM 10hrs 4571 won"<<endl;

cout<<"Select the flight you want to Book: ";
while(!(cin>>choice1))
{
    cout<<"Select the flight you want to Book: "<<endl;
    cin.clear();
    cin.ignore();
}

if(choice1==1)
{
    charges=2708;
    cout<<"\nYou have Sucessfully Booked the flight KR - 774"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==2)
{
    charges=3009;
    cout<<"\nYou have Sucessfully Booked the flight KR - 103"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==3)
{
    charges=4571;
    cout<<"\nYou have Sucessfully Booked the flight KR - 105"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
```

```
    }
    else
    {
        cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
        clearScreen();
        UserMenu();
    }

    cout<<"Press 1 to go back to Menu.."<<endl;
    cin>>back;
    system("cls");
    if(back==1)
    {

        UserMenu();
        break;
    }

    else
    {

        clearScreen();
        UserMenu();
    }

}
case 10:
{
    cout<<"_____ Welcome to Saudi
Airlines_____ \n"<<endl;
    cout<<"Your Comfort is our priority. Enjoy the journey!"<<endl;
    cout<<"Following are the flights\n"<<endl;
    cout<<"1. SV - 776"<<endl;
    cout<<"\t15-04-2024 08:00AM 10hrs 2708 Riyal"<<endl;

    cout<<"2. SV - 163"<<endl;
    cout<<"\t15-05-2024 12:00AM 11hrs 3009 Riyal"<<endl;
```

```
cout<<"3. SV - 155"<<endl;
cout<<"\t05-02-2024 07:00PM 10hrs 4571 Riyal"<<endl;

cout<<"Select the flight you want to Book: ";
while(!(cin>>choice1))
{
    cout<<"Select the flight you want to Book: "<<endl;
    cin.clear();
    cin.ignore();
}

if(choice1==1)
{
    charges=2708;
    cout<<"\nYou have Sucessfully Booked the flight SV - 776"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==2)
{
    charges=3009;
    cout<<"\nYou have Sucessfully Booked the flight SV - 163"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else if(choice1==3)
{
    charges=4571;
    cout<<"\nYou have Sucessfully Booked the flight SV - 155"<<endl;
    cout<<"You can go back to menu and take the ticket"<<endl;
    cout<<endl;
}
else
{
    cout<<"Invalid Input ,shifting to previous Menu. "<<endl;
    clearScreen();
    UserMenu();
}
```

```
        cout<<"Press 1 to go back to Menu.."<<endl;
        cin>>back;
        system("cls");
        if(back==1)
        {

            UserMenu();
            break;
        }

        else
        {

            clearScreen();
            UserMenu();
        }
    }

    default:
    {
        cout<<"Invalid Input,Shifting you to Menu";
        clearScreen();
        UserMenu();
        break;
    }
}

}

// ADMIN
// MENU OF ADMIN
int adminMenu()
{

    int optionA;
    system ("color 02");
    cout << " -----" << endl;
```

```
        cout << "| " << endl;
        cout << "|----- ADMIN MENU -----| "
<< endl;
        cout << "| " << endl;
        cout << "| " << endl;
        cout << "|----- " << endl;
        cout << endl;

        cout << "Login Menu > Login > Admin " << endl;
        cout << "-----" << endl;
        cout << "Select one of the following options..." << endl;
        cout << "1- View Passengers Data" << endl;
        cout << "2- View Passengers travel Data" << endl;
        cout << "3- Add new flights " << endl;
        cout << "4- View flights " << endl;
        cout << "5- Allot Seats " << endl;
        cout << "6- view Booked Flights " << endl;
        cout << "7- Ordered Passengers" << endl;
        cout << "8- Admin Stored data" << endl;
        cout << "9- Exit" << endl;
        cout << "Your Option..";
        while(!(cin>>optionA))
        {
            cout << "Your Option..";
            cin.clear();
            cin.ignore();
        }
        return optionA;
    }
// 1 VIEW PASSENGERS DATA
void viewPData(string pName[], string passport[], string cnic[], string mail[], string
gender[],string pNum[],string seating[],int total[],int &entry_count )
{
    cout << "Login Menu > Login > Admin > View Passengers Data" << endl;
    cout << "-----" << endl;
    cout << "Name"
        << "\t"
        << "CNIC"
        << "\t"
        << "Passport"
        << "\t"
```



```
<< "Phone"
<< "\t"
<< "Email"
<< "\t"
<< "Gender"
<< "\t"
<< "Seats"
<< "\t"
<< "Total" << endl;
for (int i = 0; i <= entry_count; i++)
{
    cout << pName[i] << "\t" << cnic[i] << "\t" << passport[i] << "\t\t" << pNum[i] <<
"\t" << mail[i] << "\t" << gender[i] << "\t" << seating[i] << "\t" << total[i] << endl;
}

}

// 2 VIEW PASSENGERS TRAVEL DATA
void viewPTravelData(string pClass[],string pDepartCity[],string pArrCity[],string
pTrip[],string pDepartDate[],int adult[],int child[],int infant[],int &entry_count )
{
    cout << "Login Menu > Login > Admin > View Passengers travel Data" << endl;
    cout << "-----" << endl;
    cout << "Departure"
        << "\t"
        << "Arrival"
        << "\t"
        << "Trip"
        << "\t"
        << "Date"
        << "\t"
        << "Adults"
        << "\t"
        << "Child"
        << "\t"
        << "Infants"
        << "\t"
        << "Class " << endl;
    for (int i = 0; i <= entry_count; i++)
    {
```

```
        cout << pDepartCity[i] << "\t\t" << pArrCity[i] << "\t" << pTrip[i] << "\t" <<
pDepartDate[i] << "\t" << adult[i] << "\t" << child[i] << "\t" << infant[i] << "\t" <<
pClass[i] << endl;
    }
    string record;
    ifstream file("PassengerTravel.txt");
    while(getline(file,record))
    {

        cout<<comma(record,0)<<"\t\t"<<comma(record,1)<<
"\t"<<comma(record,2)<<"\t"<<comma(record,3)<<"\t"<<comma(record,4)<<"\t"<<com
ma(record,5)<<"\t"<<comma(record,6)<<"\t"<<comma(record,7)<<endl;
    }
    file.close();
}
// 3 ADD FLIGHTS
void addFlights(string aDepartDate[],string aDepartCity[],string aArrCity[],string
aTrip[],string aDepartTime[],int &flight_count )
{
    cout << flight_count << endl;
    cout << "Login Menu > Login > Admin > Add new flights" << endl;
    cout << "-----" << endl;
    cout << "Departure from :- " << endl;
    getline(cin>>ws,aDepartCity[flight_count]);
    cout << "Arrival To :- " << endl;
    getline(cin>>ws,aArrCity[flight_count]);
    cout << "Trip Type :- " << endl;
    getline(cin>>ws,aTrip[flight_count]);
    cout << "Depart Date :-" << endl;
    getline(cin>>ws,aDepartDate[flight_count]);
    cout << "Depart Time :- " << endl;
    getline(cin>>ws,aDepartTime[flight_count]);

    ofstream file("admin1.txt", ios::app);
    file <<aDepartCity[flight_count] << ","<<aArrCity[flight_count]<< ","
<<aTrip[flight_count]<< "," << aDepartDate[flight_count]<< "," <<
aDepartTime[flight_count]<<endl;
    flight_count++;
    file.close();
}
```

```
}
// 4 View Flights
void viewFlightsA(string aDepartDate[],string aDepartCity[],string aArrCity[],string
aTrip[],string aDepartTime[],int &flight_count )
{
    cout << flight_count << endl;

    cout << "Login Menu > Login > Admin > View Flights" << endl;
    cout << "-----" << endl;
    cout << "Departure"
        << "\t"
        << "Arrival"
        << "\t"
        << "Trip"
        << "\t"
        << "DepartDate"
        << "\t"
        << "DepartTime"
        << endl;

    string record;
    ifstream file("admin1.txt");
    while(getline(file,record))
    {
        cout<<comma(record,0)<<"\t"<<comma(record,1)<<
"\t"<<comma(record,2)<<"\t"<<comma(record,3)<<"\t"<<comma(record,4)<<endl;
    }
    file.close();
}
// 5 ALLOT SEATS
void seats(string pClass[],string pDepartCity[],string pArrCity[],string pTrip[],string
pDepartDate[],int adult[],int child[],int infant[],string seating[],int &entry_count )
{
    int total = 0;
    string seat;
    cout << "Login Menu > Login > Admin > Allot Seats" << endl;
    cout << "-----" << endl;
    cout << "Departure"
        << "\t"
        << "Arrival"
        << "\t"
```

```
<< "Trip"
<< "\t"
<< "Date"
<< "\t"
<< "Adults"
<< "\t"
<< "Child"
<< "\t"
<< "Infants"
<< "\t"
<< "Class" << endl;
for (int i = 0; i <= entry_count; i++)
{
    cout << pDepartCity[i] << "\t" << pArrCity[i] << "\t" << pTrip[i] << "\t" <<
pDepartDate[i] << "\t" << adult[i] << "\t" << child[i] << "\t" << infant[i] << "\t" <<
pClass[i] << endl;

    cout << "No of Passengers (Adult) :- " << adult[i] << endl;
    cout << "No of Passengers (Child) :- " << child[i] << endl;
    cout << "No of Passengers (Infant) :- " << infant[i] << endl
        << endl;
    total = (adult[i]) + (child[i]);

    cout << "Enter Seat Range :- ";
    getline(cin>>ws,seating[i]);
}

}

// 6 See booked Flights
void bookedFlightsA(string usernameA[], string &username, string rolesA[],string
pDepartCity[],string pArrCity[],string pDepartDate[],int optionF[],int total[],const int
&RECORDS)
{
    int i = 0;
    cout << "Login Menu > Login > Admin > See booked Flights" << endl;
    cout << "-----" << endl;
    cout << "Your Flights..." << endl;
    for (int j = 0; j < RECORDS; j++)
    {
        if (rolesA[j] == "admin")
        {
```

```
        i--;
    }
    else if (usernameA[j] == username)
    {
        i = j + i;
        if (optionF[i] == 1)
        {
            cout << "1- From \t " << pDepartCity[i] << "\t To \t " << pArrCity[i] << " \t at
\t 12:00 \t \t " << pDepartDate[i] << endl;
            cout << "Your Total Expenditure :- " << total[i] << endl;
        }
        else if (optionF[i] == 0)
        {
            cout << "No booked flights yet!!" << endl;
        }
    }
}
}
// 7 SORTED PASSENGERS
void sorting(string pName[], string passport[], string cnic[], string mail[], string
gender[],string pNum[],string seating[],int total[],int &entry_count )
{
    cout << "Admin Menu > Login > Admin > Ordered Passengers" << endl;
    cout << "-----" << endl;
    cout << "Name"
        << "\t"
        << "CNIC"
        << "\t"
        << "Passport"
        << "\t"
        << "Phone"
        << "\t"
        << "Email"
        << "\t"
        << "Gender"
        << "\t"
        << "Seats"
        << "\t"
        << "Total" << endl;
    for (int i = 0; i <= entry_count; i++)
    {
```

```
    for (int j = i + 1; j <= entry_count; j++)
    {
        if (total[j] > total[i])
        { // sort total
            int temp = total[i];
            total[i] = total[j];
            total[j] = temp;
            // sort pName
            string namex = pName[i];
            pName[i] = pName[j];
            pName[j] = namex;
            // sort cnic array
            string name1 = cnic[i];
            cnic[i] = cnic[j];
            cnic[j] = name1;
            // sort passport
            string name2 = passport[i];
            passport[i] = passport[j];
            passport[j] = name2;
            // sort pNum
            string name3 = pNum[i];
            pNum[i] = pNum[j];
            pNum[j] = name3;
            // sort mail
            string name4 = mail[i];
            mail[i] = mail[j];
            mail[j] = name4;
            // sort gender
            string name5 = gender[i];
            gender[i] = gender[j];
            gender[j] = name5;
            // sort gender
            string name6 = seating[i];
            seating[i] = seating[j];
            seating[j] = name6;
        }
    }
}
for (int i = 0; i <= entry_count; i++)
{
```

```
        cout << pName[i] << "\t" << cnic[i] << "\t" << passport[i] << "\t\t" << pNum[i] <<
"\t" << mail[i] << "\t" << gender[i] << "\t" << seating[i] << "\t" << total[i] << endl;

    }
}

string comma(string word, int field)
{
    int commacount = 0;
    string item;
    for (int i = 0; i < word.length(); i++)
    {
        if (word[i] == ',')
        {
            commacount++;
        }
        if (commacount == field)
        {
            if (word[i] != ',')
            {
                item = item + word[i];
            }
        }
    }
    return item;
}

void Astore(string aDepartCity[],string aArrCity[],string aTrip[],string
aDepartDate[],string aDepartTime[], string pName[], string passport[], string cnic[],
string mail[], string gender[],string pNum[],string pClass[],string pDepartCity[],string
pArrCity[],string pTrip[],string pDepartDate[],int adult[],int child[],int infant[])
{
    cout << "Login Menu > Login > Passenger > Admin flights" << endl;
    cout << "-----" << endl;
    cout << "Departure"
        << "\t"
        << "Arrival"
        << "\t"
        << "Trip"
        << "\t"
        << "DepartDate"
```

```
<< "\t"
<< "DepartTime"
<< endl;

string record;
ifstream file("Admin1.txt");
while(getline(file,record))
{
    cout<<comma(record,0)<<"\t\t"<<comma(record,1)<<
"\t"<<comma(record,2)<<"\t"<<comma(record,3)<<"\t\t"<<comma(record,4)<<endl;
}
file.close();
cout<<endl;
cout << "Login Menu > Login > Admin > Passengers Data" << endl;
cout << "-----" << endl;
cout << "Name"
    << "\t"
    << "CNIC"
    << "\t"
    << "Passport"
    << "\t"
    << "Phone"
    << "\t"
    << "Email"
    << "\t"
    << "Gender"
    << endl;

ifstream file1("PassengerData.txt");
while(getline(file1,record))
{
    cout<<comma(record,0)<<"\t"<<comma(record,1)<<
"\t"<<comma(record,2)<<"\t\t"<<comma(record,3)<<"\t"<<comma(record,4)<<"\t"<<co
mma(record,5)<<endl;
}
file1.close();
cout<<endl;
cout << "Login Menu > Login > Admin > View Passengers travel Data" << endl;
cout << "-----" << endl;
cout << "Departure"
    << "\t"
```



```
<< "Arrival"
<< "\t"
<< "Trip"
<< "\t"
<< "Date"
<< "\t"
<< "Adults"
<< "\t"
<< "Child"
<< "\t"
<< "Infants"
<< "\t"
<< "Class " << endl;

ifstream file2("PassengerTravel.txt");
while(getline(file2,record))
{

    cout<<comma(record,0)<<"\t\t"<<comma(record,1)<<
"\t"<<comma(record,2)<<"\t"<<comma(record,3)<<"\t"<<comma(record,4)<<"\t"<<com
ma(record,5)<<"\t"<<comma(record,6)<<"\t"<<comma(record,7)<<endl;
}
    file2.close();

}

void credentials( string &pass, string &username, string &role)
{

    string record;
    ifstream file3("User.txt");

    while(getline(file3, record))
    {

        cout<< comma(record, 0) <<" "<<comma(record, 1) <<" "<< comma(record,
2)<<endl;

    }
    file3.close();

}
```

- **Weakness in the Business Application**

I think there are some weaknesses in the Business Application.

- There might be some good system like there must be some employee that will control user needs and owner will only see all the details
- The application structure should be good.

- **Future Directions**

My future directions are:

- I think I will make this application more advance and secure and provide more facilities to the user.
- I will make this application look beautiful.
- I will add some employee functionalities also.
- I will make the application more special.