Airline Reservation System Project Report

Table of Contents:-

1. Introduction

The Airline Reservation System is a comprehensive C++ application developed to streamline the process of booking flight tickets. This project is part of industrial training aimed at enhancing practical skills in software development. The system is designed to manage all aspects of airline ticket reservations, including booking, payment processing, seat editing, and providing allowance details. By automating these processes, the system aims to improve efficiency, accuracy, and user experience.

2. Overall Description

Project Objective:

The primary objective of this project is to create a reliable and user-friendly system that facilitates the booking of flight tickets. The system ensures seamless handling of user information, payment transactions, and seat allocation.

Project Scope:

The Airline Reservation System covers the following functionalities:

- User registration and login

- Flight search and selection

- Ticket booking and confirmation

- Payment processing

- Seat selection and editing

- Displaying baggage allowance and other details

Target Audience:

The target audience for this system includes airline customers, airline staff, and travel agents. The system is designed to cater to the needs of these users by providing a straightforward and efficient interface for managing flight reservations.

3. System Features

User Registration and Login:

- New users can register by providing personal

information, such as name, contact details, and email address:

- Registered users can log in using their credentials to access the system and manage their bookings.

Flight Search and Selection:

- Users can search for available flights based on parameters like destination, date, and class of service.

- The system displays a list of available flights that match the search criteria, including flight details like departure and arrival times, duration, and fare.

Ticket Booking and Confirmation:

- Users can select a flight from the search results and proceed to book tickets.

- The system collects necessary information such as passenger details, number of tickets, and class of service.

- Upon successful booking, a confirmation message is displayed, and a ticket is generated.

Payment Processing:

- The system supports multiple payment methods, including credit/debit cards .

- Users can securely enter their payment details, and the system processes the transaction.

- After successful payment, a receipt is generated.

Seat Selection and Editing:

- Users can select their preferred seats during the booking process.

- The system displays a seat map indicating available and occupied seats.

- Users can edit their seat selection before finalizing the booking.

Displaying Baggage Allowance and Other Details:

- The system provides information about baggage allowance for different classes of service.

- Users can view additional details such as meal preferences, special assistance, and in-flight entertainment options.

4. External Interface Requirements

User Interface:

- The system provides a graphical user interface (GUI) that is intuitive and easy to navigate.

- Forms and menus are used for user inputs and selections.

- Error messages and prompts guide users through the booking process.

Hardware Interface:

- The system requires a standard computer or laptop with a keyboard and mouse.

- An internet connection is necessary for processing payments and sending confirmation emails.

Software Interface:

- The system is developed using C++ and utilizes standard libraries for handling user inputs, file operations, and payment processing.

- It integrates with external payment gateways for secure transactions.

Communication Interface:

- The system sends email notifications to users for booking confirmations and payment receipts.

- It uses standard email protocols (SMTP) to send emails.

5. Other Requirements

Performance Requirements

- The system should respond to user actions within a few seconds.

- It should handle multiple users simultaneously without performance degradation.

Security Requirements:

- User data, including personal and payment information, must be securely stored and transmitted.

- The system should implement encryption and secure communication protocols (SSL/TLS) to protect sensitive information.

Usability Requirements:

- The system should be user-friendly, with clear instructions and feedback for user actions.

- It should support accessibility features, such as keyboard navigation and screen readers.

Reliability Requirements:

- The system should be reliable and available at all times, with minimal downtime.

- Regular backups of user data and transaction records should be maintained.

Maintainability Requirements:

- The system should be designed with modular code to facilitate easy updates and maintenance.

- Clear documentation of the code and system functionalities should be provided.

6. Conclusion

The Airline Reservation System project demonstrates the effective application of C++ programming to solve real-world problems. By automating the process of booking flight tickets, the system enhances efficiency, accuracy, and user satisfaction. The project incorporates essential features such as user registration, flight search, ticket booking, payment processing, and seat selection, making it a comprehensive solution for airline reservations. This report outlines the project's objectives, scope, features, and requirements, providing a clear understanding of the system's capabilities and benefits.

This report provides a detailed overview of the Airline Reservation System, ensuring that it covers all the necessary aspects of the project for industrial training purposes.

7.Code

- **Header Files:**

```
#include <iostream>
#include <string>
#include <conio.h>
#include <fstream>
#include <iomanip>
#include <Windows.h>
```

- **Function For Luggage Details**

```
        void LuggageDetails()
             {
        cout << "\n\n\t\t\t\t\t\t\t\tYour allowance is based on the total weight of all your
        baggage.";
        cout << "\n\n\t\t\t\t\t\t\t\tEconomy Class: ";
        cout << "\n\n\t\t\t\t\t\t\t\t2 x 25kg Baggages allowed! ";
        cout << "\n\n\t\t\t\t\t\t\t\ttBusiness Class: ";
             cout << "\n\n\t\t\t\t\t\t\t\t2x 35kg Baggages allowed! ";
             cout << "\n\n\t\t\t\t\t\t\t\tBaggage Dimensions are as follows: ";
        cout << "\n\n\t\t\t\t\t\t\t\tThe total dimensions of a bag should not exceed 300
        cm(118 inches)\n\n\n";

system("pause");
        }
```

- **Function For Payment**

```
void Payment()
{
string name, lname, cc, cvv, dob;


cout << endl
<< endl;
cout          <<          "\n\n\t\t\t\t\t\t\t\t===============PAYMENT
DETIALS===============" << endl;


cout << "\n\n\t\t\t\t\t\t\t\tEnter First Name: ";
cin >> name;
```

```
cout << "\n\n\t\t\t\t\t\t\t\tEnter Last Name: ";
cin >> lname;

check:
cout << "\n\n\t\t\t\t\t\t\t\tEnter Credit/Debit Card Number (16 digits): ";
cin >> cc;
if (cc.length() != 16)
{
cout << "\n\n\t\t\t\t\t\t\tCC number must have a length of 16 only! \n";
goto check;
}
else
check2:
cout << "\n\n\t\t\t\t\t\t\t\tEnter CVV (3 digits): ";
for (int i = 0; i > -1; i++)
{
char temp;
temp = _getch();
if (temp != 13 && temp != 8)
{
_putch('*');
}
if (temp == 13)
{
break;
}
if (temp == 8 || temp == 127 && !cvv.empty())
{
cout << "\b \b";
cvv.erase(cvv.size() - 1);
}
else
cvv += temp;
}
if (cvv.length() != 3)
{
cout << "\n\n\t\t\t\t\t\t\tCVV number must have a length of 3 only! \n";
cvv = "";
goto check2;
}
else
{
cout << "\n\n\t\t\t\t\t\t\t\tEnter Date Of Birth(DD/MM/YYYY): ";
cin >> dob;
cout << "\n\n\t\t\t\t\t\t\t\tYour payment is successfully processed!";

fstream payfile;
payfile.open("Payment_Details.txt", payfile.app);
payfile << name << "-" << lname << "-" << cc << "-" << cvv << "-" << dob << "-"
<< "\n";
payfile.close();
}
cout << "\n\n\t\t\t\t\t\t\t\t";
```

```
system("pause");
}
```

- **Function for seat book**

```
string SeatChoose()
{
fstream Seat;
int count = 0, delimit = 0;
string line;
SeatRecord Seats[10];
Seat.open("Seat_Details.txt");
while (getline(Seat, line))
{
count++;
}
Seat.close();
Seat.open("Seat_Details.txt");
for (int j = 0; j < count; j++)
{
string line1;
getline(Seat, line1);
for (int i = 0; i > -1; i++)
{
char temp;
temp = line1[i];
if (temp == '-')
{
delimit = i;
break;
}
Seats[j].RowA += temp;
}

for (int i = delimit + 1; i > -1; i++)
{
char temp;
temp = line1[i];
if (temp == '-')
{
delimit = i;
break;
}
Seats[j].RowB += temp;
}

for (int i = delimit + 1; i > -1; i++)
{
char temp;
temp = line1[i];
if (temp == '-')
{
delimit = i;
break;
}
Seats[j].RowC += temp;
}
}
```

- **Function for display ticket**

```
void Display(int c)
{
int Counter;
Counter = SplitStruct();
system("cls");
char choice;
do
{
system("cls");
system("color 11");
cout << "\n\n\t\t\t\t\t\t\t         Airline Ticket Management System\n";
cout << "\n\n\t\t\t\t\t\t\t       ============= MAIN MENU =============\n";
cout << "\n\n\t\t\t\t\t\t\t\t01. Create New Ticket.\n";
cout << "\n\n\t\t\t\t\t\t\t\t02. Edit Reserved Tickets.\n";
cout << "\n\n\t\t\t\t\t\t\t\t03. Delete Your Ticket.\n";
cout << "\n\n\t\t\t\t\t\t\t\t04. Print/Display Booked Ticket.\n";
cout << "\n\n\t\t\t\t\t\t\t\t05. Luggage Allowance Details.\n";
cout << "\n\n\t\t\t\t\t\t\t\t06 Exit.\n";
cout << "\n\n\t\t\t\t\t\t\t\tSelect Your Option (1 - 6): ";
cin >> choice;
system("cls");

if (choice == '1')
{
NewTicket(c);
Counter++;
}
else if (choice == '2')
{
EditTicket();
}
else if (choice == '3')
{
DelTicket();
}
else if (choice == '4')
{
printTic(Counter);
}
else if (choice == '5')
{
LuggageDetails();
}
else if (choice == '6')
{
cout << "\n\n\t\t\t\t\t\t\t\tThe program will now exit!";
cout << "\n\n\t\t\t\t\t\t\t\t";
break;
}

} while (choice != '6');
}
```

- **Function for login**

```
void Login(int c)
{
system("cls");
User arr[200];
int delimit = 0, count = 0, Iterator = 0;
file.open("Login_Details.txt");
string line;
```

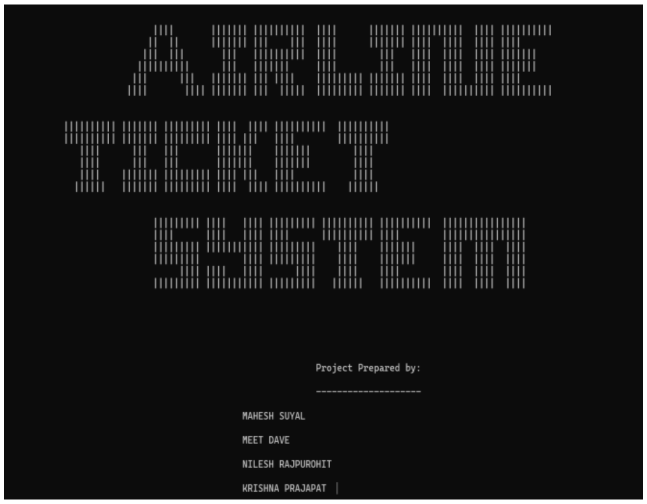
```
while (getline(file, line)) // dynamic
{
count++;
}
file.close();
file.open("Login_Details.txt");

for (int j = 0; j < count; j++) // dynamic needed here
{
string line;
getline(file, line);
for (int i = 0; i > -1; i++)
{
char temp;
temp = line[i];
if (temp == '-')
{
delimit = i;
break;
}
arr[j].username += temp;
}

for (int i = delimit + 1; i > -1; i++)
{
char temp;
temp = line[i];
if (temp == '-')
{
delimit = i;
break;
}
```

```
arr[j].password += temp;
}
}

cout << "\n\n\t\t\t\t\t\t\t ==================== LOGIN
==================== \n\n";
string user, pass;
bool login = false;

// checking username and password:
check:
cout << "\n\n\t\t\t\t\t\t\tEnter authorized Username: ";
getchar();
getline(cin, user);
cout << "\n\n\t\t\t\t\t\t\tEnter authorized Password: ";

for (int i = 0; i > -1; i++)
{
char temp;
temp = _getch();
if (temp != 13 && temp != 8)
{
_putch('*');
}
if (temp == 13)
{
break;
}
if (temp == 8 || temp == 127 && !pass.empty())
{
cout << "\b \b";
pass.erase(pass.size() - 1);
```

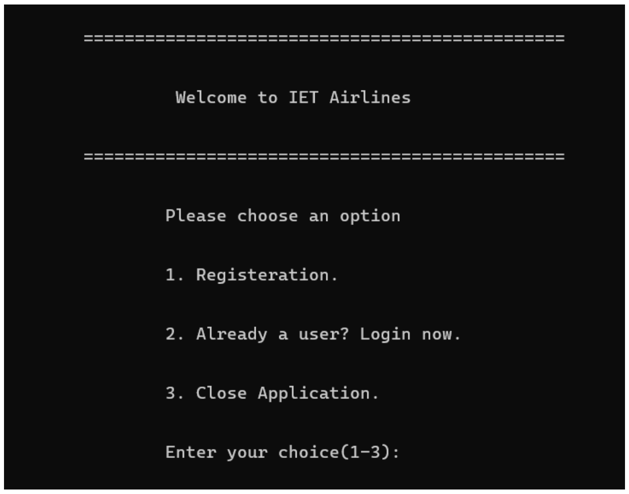
```
}
else
pass += temp;
}
cout << "\n";

for (int i = 0; i < count; i++) // and here
{
if (user == arr[i].username && pass == arr[i].password)
{
login = true;
break;
}
}

if (login == true)
{
cout << "\n\n\t\t\t\t\t\t\t\t\tLogged in! \n";
cout << "\n\n\t\t\t\t\t\t\t\tSessional serial number is: " << c << "\n\n\t\t\t\t\t\t\t\t\t";
cout << "\n\n\t\t\t\t\t\t\t\t\tLoading";
Sleep(200);
cout << ".";
Sleep(200);
cout << ".";
Sleep(400);
cout << ".";
Sleep(200);
cout << ".";
Sleep(200);
cout << ".";
Sleep(200);
Sleep(2000);
```

```
Display(c);
}
else
{
Beep(1000, 600);
cout << "\n\n\t\t\t\t\t\t\tInvalid login credentials, please try again! \n";
user = "";
pass = "";
goto check;
}

file.close();
}
```

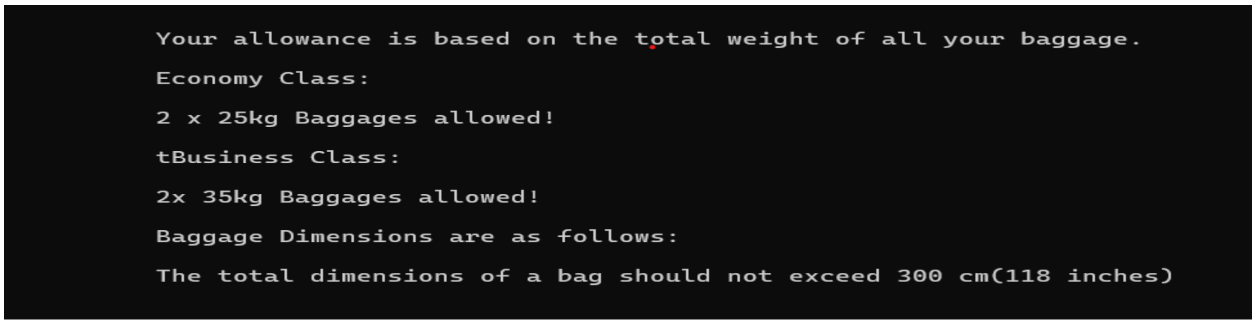
- ❖ Snapshots



```
                    Project Prepared by:
                    --------------------
         MAHESH SUYAL
         MEET DAVE
         NILESH RAJPUROHIT
         KRISHNA PRAJAPAT
```



```
===============================================

           Welcome to IET Airlines

===============================================

          Please choose an option

          1. Registeration.

          2. Already a user? Login now.

          3. Close Application.

          Enter your choice(1-3):
```



```
Your allowance is based on the total weight of all your baggage.
Economy Class:
2 x 25kg Baggages allowed!
tBusiness Class:
2x 35kg Baggages allowed!
Baggage Dimensions are as follows:
The total dimensions of a bag should not exceed 300 cm(118 inches)
```