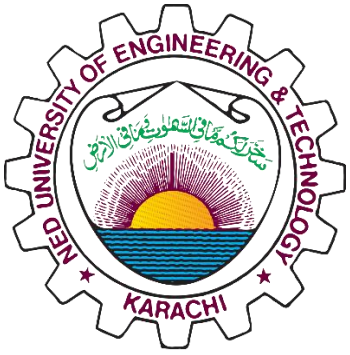
NED University of Engineering and Technology



Flight Reservation

Project Report

Programming Fundamentals (CT-175)

Group Members:

Muhammad Abdullah Ayub CTCY-042

Muhammad Azhan Javed CTCY-046

Muhammad Hasan Khan CTCY-047

Introduction:

In today's fast-paced world, efficient and user-friendly flight reservation methods are crucial for both travelers and airlines. The Flight Reservation Project presented here is designed to simplify the booking and management of flight tickets. This system provides a seamless experience for users to check available flights, book tickets, cancel reservations, and track the progress of their flights.

Product Features:

Show Available Flights:

Users can view a list of available flights based on their specified criteria such as date, destination, and departure location.

Information includes flight number, departure time, arrival time, and available seats.

PF PROJECT.c

```c
    {
    system("cls");
    printf("\n\n\t\t\t\t********************** CYBER Airlines ***********************");
    printf("\n\n\n\n\n\n\t\t\t\t\t1. Show available Flights.\n");
    printf("\n\t\t\t\t\t2. Book a ticket\n");
    printf("\n\t\t\t\t\t3. Cancel Ticket.\n");
    printf("\n\t\t\t\t\t4. Check your Flight Progress.\n");
    printf("\n\t\t\t\t\t5. Exit.\n\n");
    printf("\t\t\t\t\tEnter your Choice(1-5): ");
    scanf("%d",&choice);

    switch(choice)
    {
        case 1:
        {
            available_flights();
            break;
        }
        case 2:
        {
            booking();
            break;
        }
        case 3:
        {
            cancel();
            break;
        }
        case 4:
        {
            progress();
            break;
```

Book a Ticket:

Users can easily book a ticket by providing necessary details such as passenger information and preferred seat selection.

The system validates seat availability and updates the database accordingly.

PF PROJECT.c

```c
int booking()
{

    do
    {
        system("cls");
        printf("\n\n\t\t\t\t********************** BOOKING **********************");
        printf("\n\n\n\n\n\n\t\t\t\t1. Bussiness Class.(Rs.45000)\n");
        printf("\n\t\t\t\t\t2. Economy Class.(Rs.30000)\n");
        printf("\n\t\t\t\t\t3. Exit.\n\n");
        printf("\t\t\t\t\tEnter your Choice(1-3): ");
        scanf("%d",&c);

        switch(c)
        {
            case 1:
            {
                system("cls");
                printf("\n\n\t\t\t\t********************** BOOKING **********************");
                flights();
                printf("\n\n\n\n\t\t\tEnter Flight Number(101-105): ");
                scanf("%d",&flightnun);

                switch(flightnun)
                {

                case 101:
                {
                printf("\n\n\n\n\t\t\tWould you like to see the Seating Pattern for Bussiness class?\n");
                printf("\n\t\t\t\t\tPress 1 if Yes and 0 if No: ");
                scanf("%d",&c2);


                if(c2 == 1)
                {
                    k=1;
                    system("cls");
                    printf("\n\n\t\t\t\t********************** Seating Pattern **********************\n\n");
                    printf("\n\n\n");
                    for(i=1;i<=7;i++)
                    {
                        for(j=1;j<=3;j++)
                        {
                            printf("\t\tSeat%d\t\t",k++);
                        }
                        printf("\n\n\n");
                    }
                    printf("\n\n\t\t\t\t\tPress any key to Continue...");
                    getch();
                }

                do
                {
                    system("cls");
                    printf("\n\n\t\t\t\t********************** BOOKING **********************");
                    printf("\n\n\n\n\n\t\t\t\tAvailable Seats are (1-21)\n\n");
                    printf("\t\t\t\tEnter your Seat Number: ");
                    scanf("%d",&snun);

                    if(seats_101[snun-1] == 1)
                    {
                        system("cls");
                        printf("\n\n\t\t\t\t********************** BOOKING **********************");
                        printf("\n\n\n\n\n\t\t\tSeat Taken Would you like to choose another seat?");
```

Cancel Ticket:

Users can cancel their reservations with minimal effort.

Cancellation logic ensures accurate seat availability updates in real-time.

PF PROJECT.c

```c
int cancel()
{
    int fn, sn;

    system("cls");
    printf("\n\n\t\t\t\t********************** CANCELLATION ***********************");
    printf("\n\n\n\n\t\t\t\t\tEnter Your Flight Number(101-105): ");
    scanf("%d",&fn);
    printf("\n\n\t\t\t\t\tEnter Your Seat Number: ");
    scanf("%d",&sn);
    switch(fn)
    {
        case 101:
        {
            system("cls");
            printf("\n\n\t\t\t\t********************** CANCELLATION ***********************");
            if(seats_101[sn-1]==1)
            {
                seats_101[sn-1] = 0;
                printf("\n\n\n\n\n\n\t\t\t\t*****************************************************");
                printf("\n\n\n\n\n\n\t\t\t\t\tCANCELLATION SUCCESSFULL\n");
                printf("\n\n\n\n\n\t\t\t\t*****************************************************");
                getch();
                break;
            }
            else
            {
                printf("\n\n\n\n\n\n\n\t\t\t\t*****************************************************");
                printf("\n\n\n\n\n\n\t\t\t\t\tYOU DON'T HAVE A RESERVATION\n");
                printf("\n\n\n\n\n\t\t\t\t*****************************************************");
                getch();
                break;
            }
        }
        case 102:
        {
            system("cls");
            printf("\n\n\t\t\t\t********************** CANCELLATION ***********************");
            if(seats_102[sn-1]==1)
            {
                seats_102 [sn-1] = 0;
                printf("\n\n\n\n\n\n\n\t\t\t\t*****************************************************");
                printf("\n\n\n\n\n\n\t\t\t\t\tCANCELLATION SUCCESSFULL\n");
                printf("\n\n\n\n\n\t\t\t\t*****************************************************");
                getch();
```

Check Your Flight Progress:

Passengers can track the progress of their booked flights.

Information includes current location, estimated arrival time, and any delays if applicable.

```c
int progress()
{
    int num, fn;

    system("cls");
    num = rand();

    printf("\n\n\t\t\t\t********************** FLIGHT PROGRESS ***********************");
    printf("\n\n\n\n\t\t\t\t\tEnter Your Flight Number(101-105): ");
    scanf("%d",&fn);

    system("cls");

    if(fn < 101 || fn > 105 )
    {
        printf("\n\n\t\t\t\t********************** FLIGHT PROGRESS ***********************");
        printf("\n\n\n\n\n\t\t\t\t\tInvalid Flight Number Entered.....");
    }
    else if(num%2 == 0)
    {
        printf("\n\n\t\t\t\t********************** FLIGHT PROGRESS ***********************");
        printf("\n\n\n\n\n\n\n\t\t----------------------------------------------------------------------------");
        printf("\n\n\n\t\t\tFLIGHT WILL BE ARRIVING SHORTLY");
        printf("\n\n\n\t\t----------------------------------------------------------------------------");
    }
    else
    {
        printf("\n\n\t\t\t\t********************** FLIGHT PROGRESS ***********************");
        printf("\n\n\n\n\n\n\n\n\t\t----------------------------------------------------------------------------");
        printf("\n\n\n\t\t\t  FLIGHT WILL BE DELAYED DUE TO SOME ISSUE");
        printf("\n\n\n\t\t----------------------------------------------------------------------------");
    }

    getch();

}
```

Exit:

Provides a smooth and user-friendly exit option for users to conclude their interaction with the system.

```
100         case 5:
101         {
102             t = total(countbus,counteco);
103             system("cls");
104             printf("\n\n\n\n\n\n\n\t\t\t\t**************************************************");
105             printf("\n\n\n\n\n\n\t\t\t\tYour Total Fare for all tickets is: %d\n",t);
106             printf("\n\n\n\n\n\t\t\t\t**************************************************");
107             getch();
108             system("cls");
109             printf("\n\n\n\n\n\n\n\t\t\t\t**************************************************");
110             printf("\n\n\n\n\n\n\t\t\t\tThank You for using our Services.\n");
111             printf("\n\n\n\n\n\t\t\t\t**************************************************");
112             getch();
113 
114             break;
115         }
116         default:
117         {
118             printf("\n\n\t\t\t\t\tInvalid Choice.....");
119         }
120     }
121     }
```

Project Specification:

The program will show a menu with the options; Show available flights, Book a flight, Cancel a ticket, Flight progress and Exit.

By choosing any of the options, the procedure or function will be called and performs tasks accordingly.

The program will print a ticket if the booking was successful and calculated and display the total fare of the customer as he exits the program.

Solution Design:

The program requires a login by a specific username and password and then proceed to the main function.

Our project consists of 4 major modules which are:

Booking:

This module books a ticket for the customer for either business or economy class for the desired seat of customer by first showing the flights and then the seating pattern for both business and economy and then also prints a ticket for the customer if the booking is successful by using a user-defined function “ticket”.

Available Flights:

This module shows the available flights to the customers and then ask them if they want to book a ticket for any of the available flights.

Cancel:

This module cancels a reservation by asking the flights number and seat number from the customer.

Flight Progress:

As at this point we cannot know if any flight will be delayed or will be on time so we made this module as an assumption and used random function to generate a number and then decide if the flight is delayed or not on the basis of the number being odd or even.

Apart from these we have the main function which call these modules as per user’s choice and then as the user chooses to exit

the program it then calls the function “total” to calculate the total fare and then display it.

Implementation & Testing:

We tested every module and function individually at first (Alpha testing) and then when they were working properly, we combined then gradually and tested them with one another (Integrated testing) and finally combined all our modules, procedures and functions and performed final testing and checked by entering normal data, abnormal data and boundary data.

Project Breakdown Structure:

There was no workload distribution as we live nearby, so we made this project together by equally dividing the typing and thinking processes and managed to complete.

- Booking Module. (With testing)
- Available Flights, main function and login. (With testing)

- Cancel, Flight Progress and minor functions such as ticket (final ticket printing function) and total (calculation total fare). (With testing)

- Integrated testing of Procedures and Functions.

- Final testing and covering any areas of lacking.

Results:

We were able to produce the program that we decided although it has some lacking but still we managed to make a program that we proposed off and worthy of being examined.

Conclusion:

This is a complete flight reservation project of five flights of 42 seats (21 each for business and economy). It validates that a seat is not booked more than once, prints a ticket, calculates total fare, and protects information through login procedure.

Presented here is the robust solution that addresses the key requirements of a modern flight booking

platform. It offers a user-friendly interface for passengers to efficiently check, book, and manage their flights.

This project serves as a foundation for further enhancements and integration with additional features to meet evolving industry needs.