KLE Society's

KLE Technological University



**Open Ended Activity Report**

On

**Airline Reservation and Management System**

Object Oriented Programming with C++ (19ECSC204)

Object Oriented Programming with C++ Lab (18ECSP203)

**Submitted by**

Name Roll No. USN

Dhanashri Prakash Rane 108 01FE19BCS010

Vighnesh Nagesh Pai 110 01FE19BCS013

Prerana P Naik 131 01FE19BCS036

Himanshu Kumar Purbey 136 01FE19BCS041

**Team Number**: 4A07

**Faculty In-charge**:

Proff. Somashekar Patil

SCHOOL OF COMPUTER SCIENCE & ENGINEERING

HUBLI – 580 031 (India)

Academic year 2020-21

1. **Introduction**
   1. Overview of the Problem Statement
   2. Features of Application
      1. Airline Management System
      2. Airline Reservation System

**2. Design**

2.1 Class Diagram

* 1. Description of each Class
  2. Main Function
  3. Use of Standard Design Patterns

1. **Unit Test Plan**

**4. Implementation**

1. **Introduction**
   1. **Overview on Problem Statement**

The Indian Airline Management and Reservation System has two main divisions, Management System and Reservation System. Management System consists of pilots, co-pilots and crew. The Reservation System consists of passengers and customers. Each of these systems have their own functions. In Management System, the employees have schedule, flying hours, etc. In Reservation System, the customer can book seats, cancel seats, reschedule flights, etc. Our application will provide information about the airline system and its network.

* 1. **Features of Application**
     1. **Airline Management System**

A person can log in by providing his/her basic details like username and password. The details of the person are checked for authenticity in the database. If the persons’ details are authentic, they can access the Management System, otherwise the access is denied.

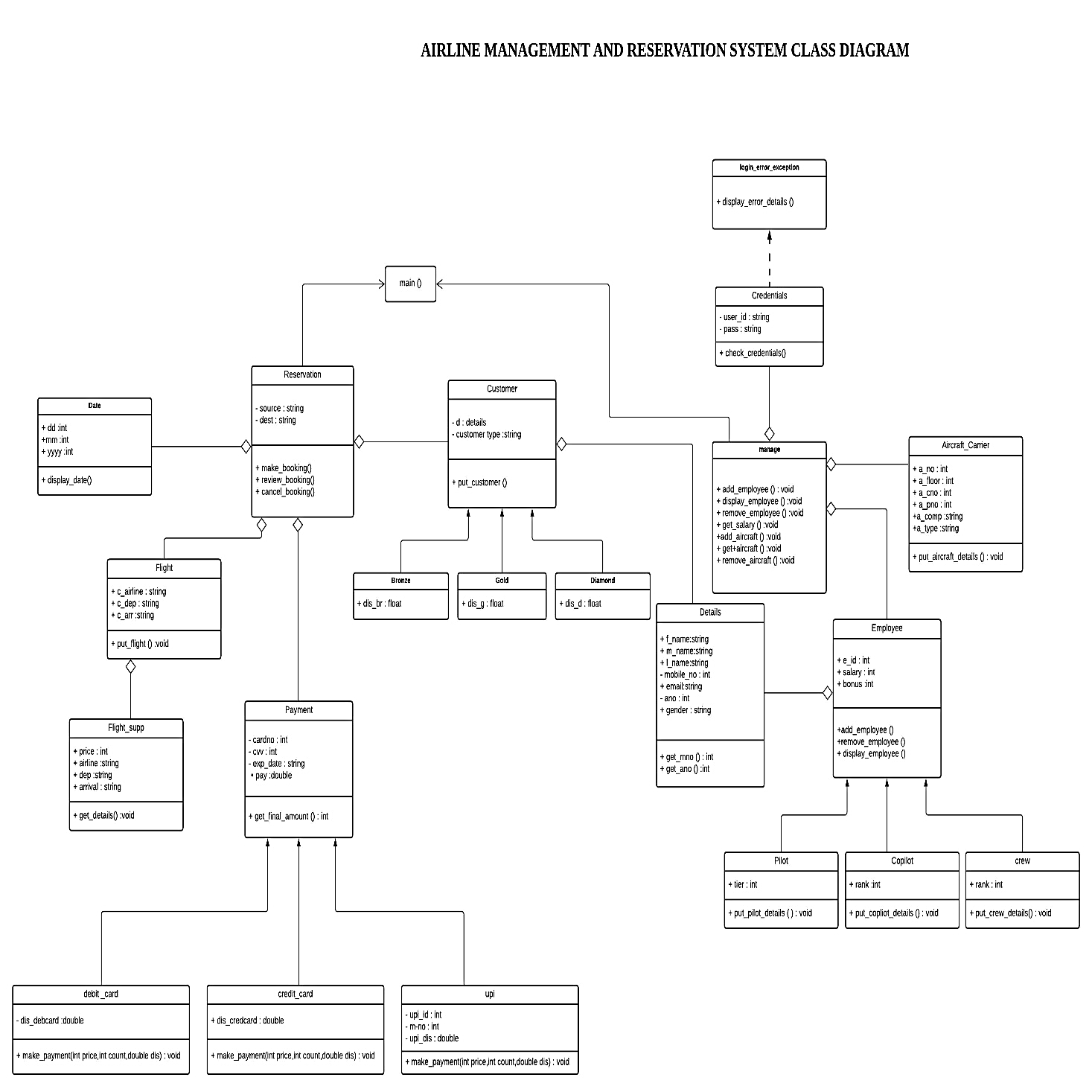
The employee can be removed from the schedule.

* + 1. **Airline Reservation System**

The customer can opt to make, review and cancel booking. The source and destination of the flight can be selected. After entering the desired date, different airlines with their departure time, arrival time and cost is

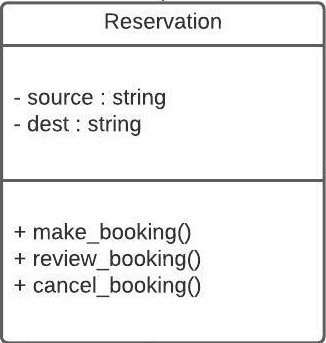
displayed. An option to enter n number of passengers and to choose if you have any membership. Membership has the advantage of discounts on flights.

1. **Design**

**2.1 Class Diagram**

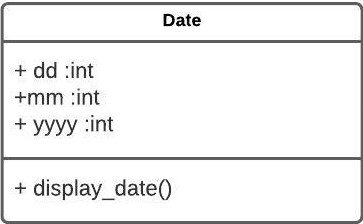
**2.2 Description of each class**

1. Reservation

****

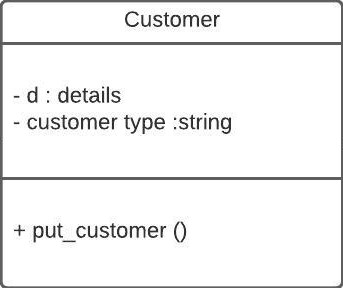
* This class has 'has a relation' with class Date, class Flight class Payment and class Customer.
* This class has make booking, review booking and cancel booking functions.

2. Date



* This class is used to display dates.
* Reservation class has class Date.

1. Customer



* This class has 'has a relation' with class Details.
* This parent class has three child classes.
* The three-child class are Bronze class, Gold class and Diamond class.

1. Bronze



* This child class is used to give discounts.

1. Gold



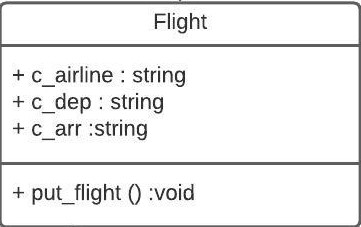
* This child class is used to give discounts.

1. Diamond



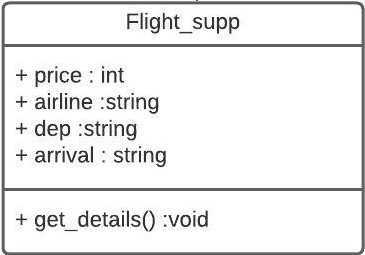
* This child class is used to give discounts.

1. Flight



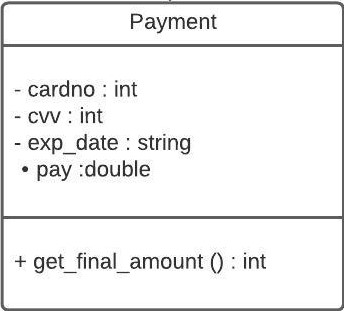
* This class has 'has a relation' with class Flight\_supp.
* This class contains departure and arrival time.
* It has object of class Flight\_supp.
* This is a Creational Design Pattern.

1. Flight\_supp



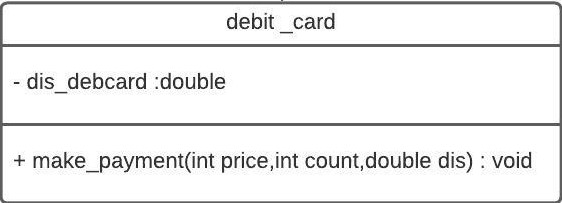
* This class is used for airline and price.
* Class Flight has Flight\_supp.

1. Payment



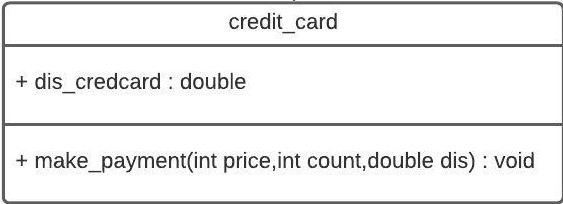
* This parent class has three child classes.
* The three child classes are class Debit\_card, class Credit\_card and class upi.

10. Debit\_card



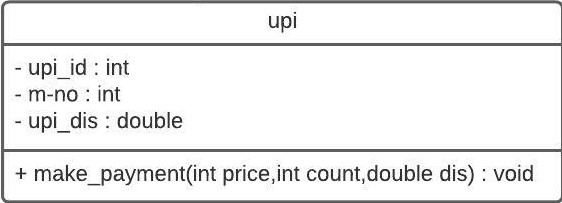
* This is a child class used to make payments.
* The make\_payment() function is overridden.

11. Credit\_card



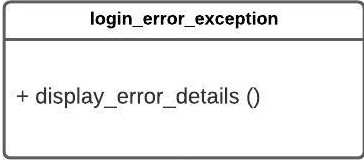
* This is a child class used to make payments.
* The make\_payment() function is overridden.

12. UPI



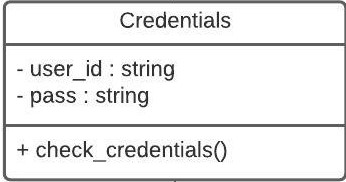
* This is a child class used to make payments.
* The make\_payment() function is overridden.

13. Login\_error\_exception



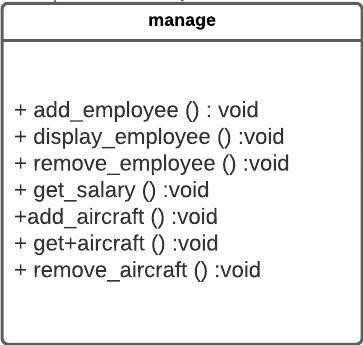
* If the username and password is not in the database and is invalid, this will throw an exception.

14. Credentials



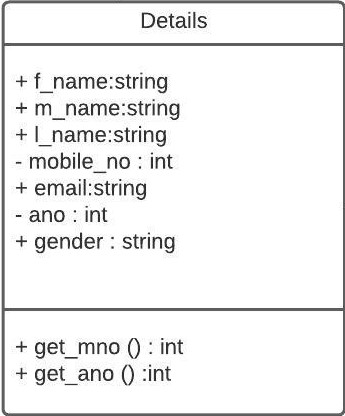
* This class has variables userid and password.
* Its function is to check credentials.

15. Manage



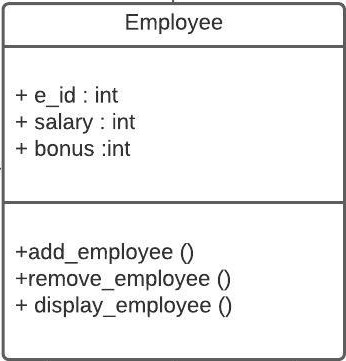
* This class is used to add, remove, display employees.

16. Details



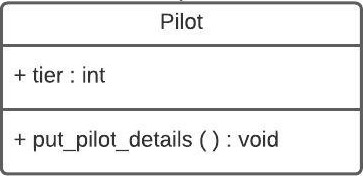
* This class contains first name, middle name, last name, mobile number, email and gender.

17. Employee



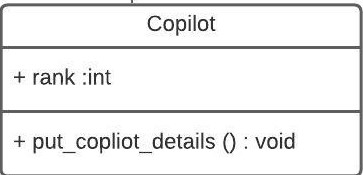
* This parent class has three child classes.
* The three child classes are pilot, copilot and crew.
* The add\_employee(), remove\_employee() and display\_employee is overridden from class Manage.

18. Pilot



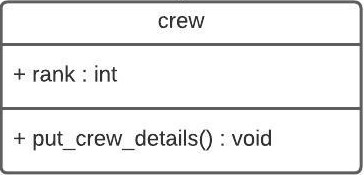
* This is a child class used for plot details.

19. Copilot



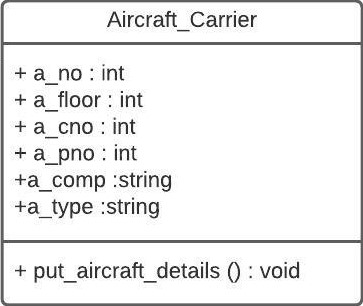
* This is a child class used for plot details.

20. Crew



* This is a child class used for crew details.

21. Aircraft\_Carrier



* This class has the aircraft details.

**2.3 Main Function**

There is a switch menu for different options to select Reservation or Management

According to user input, if reservation is selected, we will make bookings, review bookings and cancel bookings. Select source for boarding flight and destination. Enter date and select suitable flight. Enter number of travellers, if you have any membership, discount is given. Customer details should be entered. Payment mode should be selected. The booking is done. The reservation can be reviewed or cancelled by entering mobile number.

If management is selected, user name and password must be entered. Employee can be added removed. Aircraft can be added or removed. For adding employee, appropriate employee details must be entered. To remove an employee, enter the employee id. Appropriate details to be added to add or remove aircraft.

**2.4 Use of Standard Design Patterns**

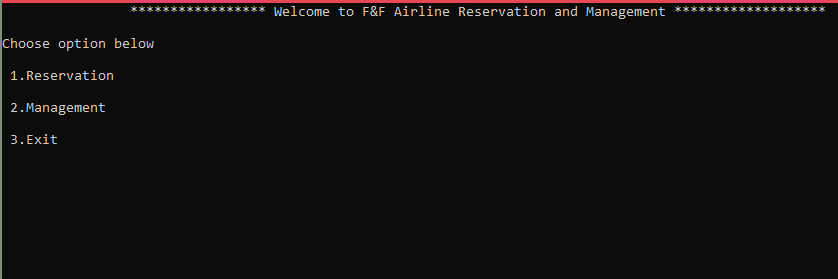
Factory method pattern (creational pattern):

* Definition: - The Factory Method pattern is a design pattern used to define a runtime interface for creating an object. It’s called a factory because it creates various types of objects without necessarily knowing what kind of object it creates or how to create it.
* Usage: - Factory method is suitable for this scenario because there any many objects which is needed to be created as per user demand (Type of customer, type of payment type of employee) these all are decided by the user at run time and if required its object is created.

**3. Unit Test Plan**

**1. Reservation**

The airline management system involves initial 3 stages of making reservation, management, and exiting the portal respectively.



**Test case: Reservation section**

**Input:**

\*\*\*\*\*\*\*\*\*\* Welcome to \_\_ Airline Reservation and Management \*\*\*\*\*\*\*\*\*\*

Choose option below

1.Reservation

2.Management

3.Exit

1

**Expected Output:**

Choose option below

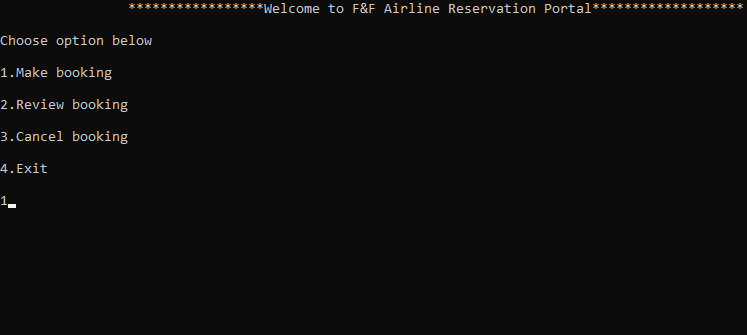
1.Make booking

2.Review booking

3.Cancel booking

4.Exit

**Actual Output:**



**Test Case: Making the booking**

**Input:**

Choose option below

1.Make booking

2.Review booking

3.Cancel booking

4.Exit

1

**Expected Output:**

Choose source for boarding flight

1.Delhi(DEL)

2.Mumbai(BOM)

3.Bengaluru (BLR)

4.Patna(PAT)

5.Kolkata(CCU)

6.Hyderabad(HYD)

7.Chennai(MAA)

8.Goa

3

Choose destination

1.Delhi(DEL)

2.Mumbai(BOM)

3.Bengaluru (BLR)

4.Patna(PAT)

5.Kolkata(CCU)

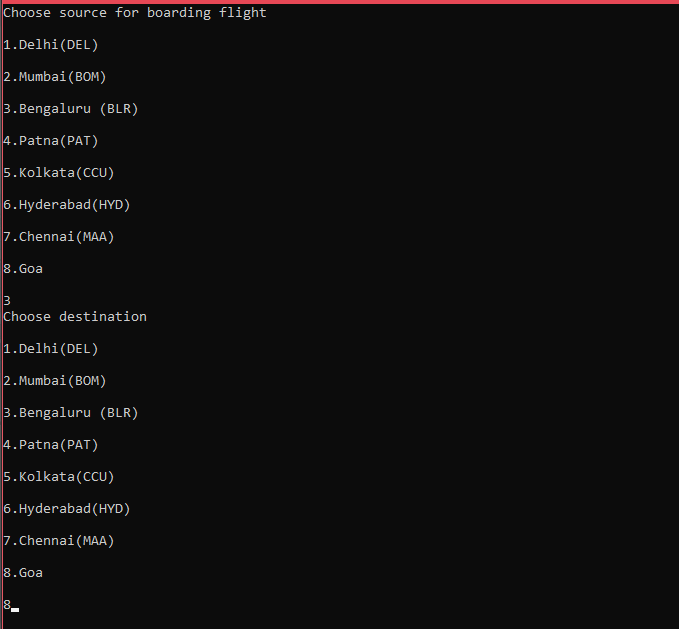
6.Hyderabad(HYD)

7.Chennai(MAA)

8.Goa

8

**Actual Output:**



**Test case: Entering Credentials and Payment Portal**

**Input:**

Enter date (DD/MM/YYYY)

12

05

2021

Choose from below

1 Indigo 09:15 (Dep) 11:20 (Arrival) rs.4625

2 Vistra 05:15 (Dep) 07:10 (Arrival) rs.8945

3 AirAsia 08:00(Dep) 11:30(Arrival) rs.8640

4 Spicejet 03:30(Dep) 05:00(Arrival) rs.6700

5 Kingfisher 07:15(Dep) 09:30(Arrival) rs.1455

1

Enter number of travelers booking is to be made

1

Choose your membership

1.Diamond

2.Gold

3.Bronze

4.No membership

2

Enter your first name

Bunty

Enter your middle name

Ramu

Enter your last name

Pandey

Enter your mobile number

8411030856

Enter your email id

xyz@gmail.com

Enter your gender

Male

Enter your Aadhar number

456673214

**Expected Output:**

Redirecting to payment portal .........

Press any key to continue . . .

1.Debit card

2.Credit Card

3.Upi

1

Enter card details

Enter debit card number

254332

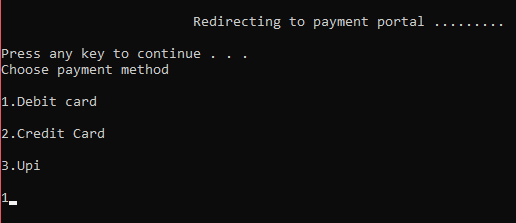
Enter exp date(mm/yy)

04/24

Enter cvv

307

**Actual Output:**



**Test Case: Payment confirmation and Detail display**

**Output:**

Processing your Payment.......

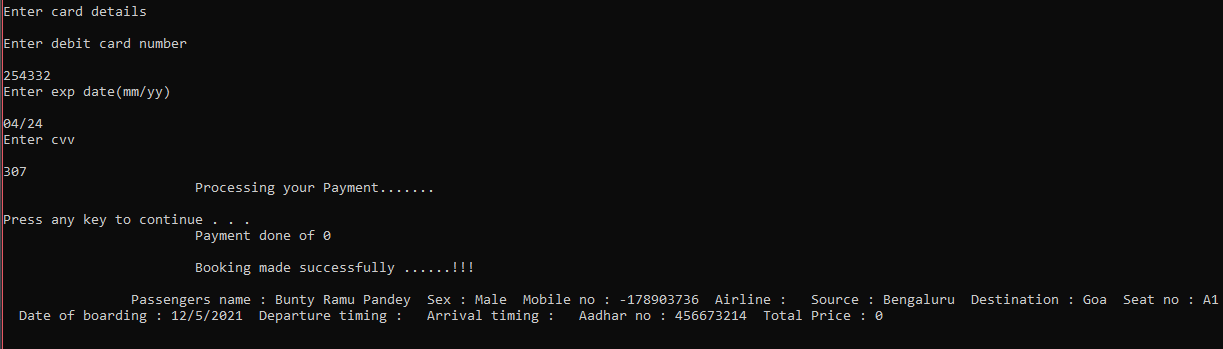
Press any key to continue . . .

Payment done of 3931.25

Booking made successfully ......!!!

Passenger’s name: Bunty Ramu Pandey Sex: Male Mobile no: 8411030856 Airline: Indigo Source: Delhi Destination: Patna Seat no: A1 Date of boarding: 12/5/2021 Departure timing: 09:15 Arrival timing: 11:20 Aadhar no: 456673214 Total Price: 3931

**Actual Output:**



**Test Case: Reviewing a Booking**

**Input:**

Choose option below

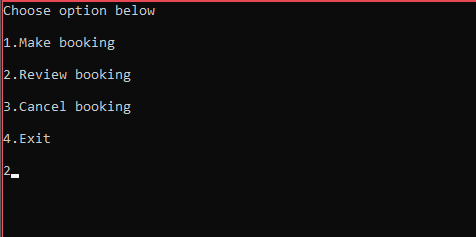
1.Make booking

2.Review booking

3.Cancel booking

4.Exit

2



**Expected Input:**

Enter your mobile no to review your bookings

8411030856

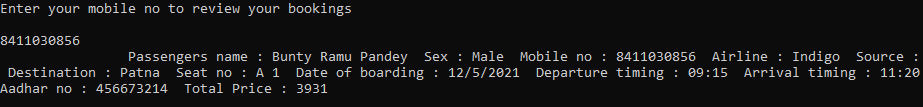
**Expected Output:**

Enter your mobile no to review your bookings

8411030856

Passenger’s name: Bunty Ramu Pandey Sex: Male Mobile no: 8411030856 Airline: Indigo Source: Destination: Patna Seat no: A 1 Date of boarding: 12/5/2021 Departure timing: 09:15 Arrival timing: 11:20 Aadhar no: 456673214 Total Price: 3931

**Actual Output:**



**Test Case: Cancelling the Booking**

**Input:**

Choose option below

1.Make booking

2.Review booking

3.Cancel booking

4.Exit

3

**Expected Input:**

Enter mobile no to cancel booking

8411030856

**Expected Output:**

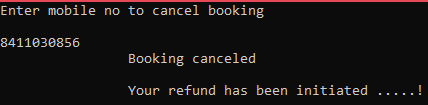
Enter mobile no to cancel booking

8411030856

Booking canceled

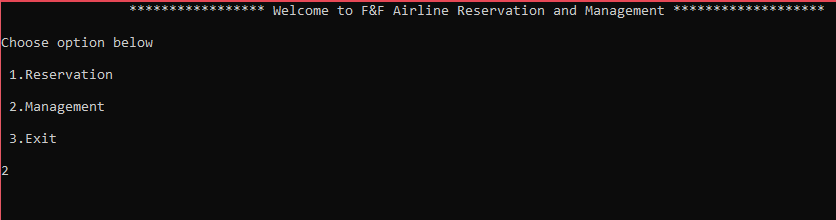
Your refund has been initiated ......!

**Actual Output:**



**2. Management**

The second stage in in this system would be management which would be only accessible by authorized personals and would include all facilities such as adding employee, adding aircrafts, getting salary etc.



**Test Case: Login with username and password of Authorized personals, return false if Incorrect.**

**Exception Handling**

\*Correct username and passwords are as follows:



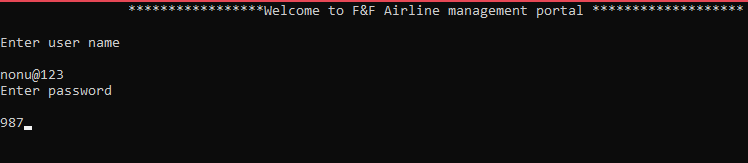
**Input: (Correct)**

Enter user name

nonu@123

Enter password

987



**Expected Output:**

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

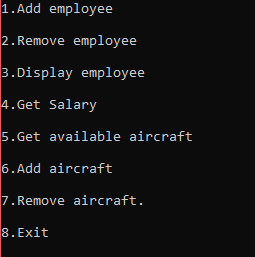
5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

**Actual Output:**



**Test Case: Adding Employee**

**Input:**

Enter employee type (Pilot/Copilot/Crew)

Pilot

Enter employee first name

Selmon

Enter employee middle name

Bhai

Enter employee last name

Khan

Enter employee gender

Male

Enter employee id

12566

Enter employee mobile number

9834573210

Enter employee base salary

15000

Enter tier of Pilot

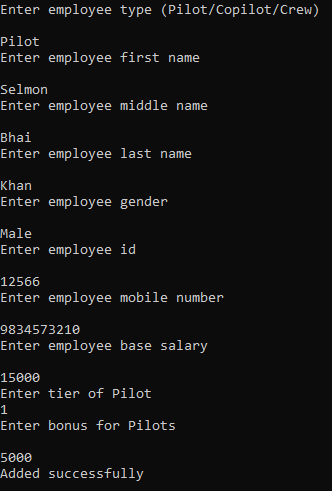
1

Enter bonus for Pilots

5000

Added successfully

**Output:**



**Test Case: Displaying Employee**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

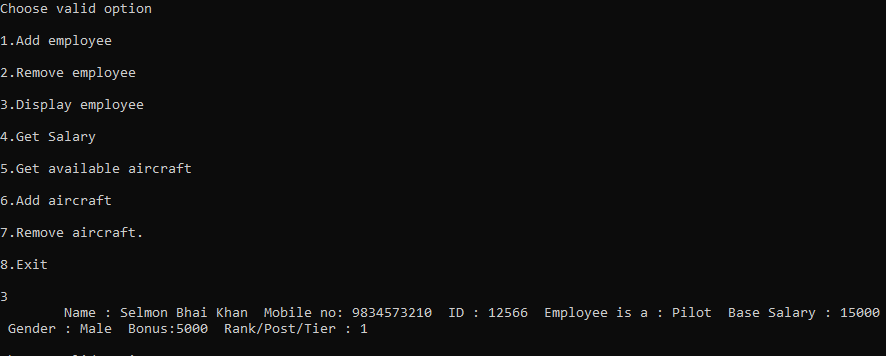
8.Exit

3

**Expected Output:**

Name: Selman Bhai Khan Mobile no: 9834573210 ID: 12566 Employee is a: Pilot Base Salary: 15000 Gender: Male Bonus:5000 Rank/Post/Tier: 1

**Actual Output:**



**Test case: Removing Employee**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

2

Enter employee id

12566

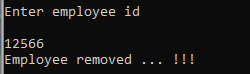
**Expected Output (When employee found):**

Enter employee id

12566

Employee removed ...!!!

**Actual Output (When employee found):**



**Test Case: Getting Salary Information**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

4

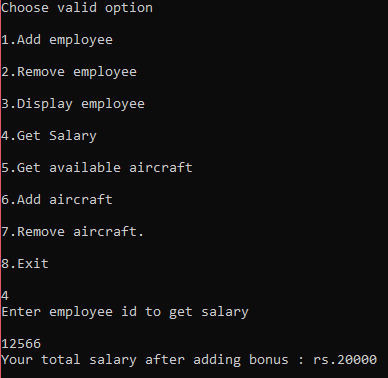
**Expected Output: (When data is available)**

Enter employee id to get salary

12566

Your total salary after adding bonus: rs.20000

**Actual Output:**



**Test Case: Add Aircrafts**

**Input:**

Enter aircraft type

Passenger

Enter aircraft no

1

Enter aircraft company

Spicejet

Enter no of decks in aircraft

2

Enter no of crew(Pilot/Co Pilot/Crew) needed for operation

25

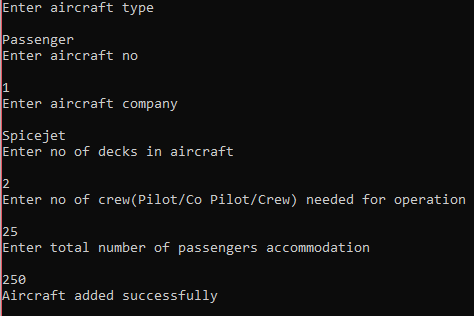
Enter total number of passengers accommodation

250

**Expected Output:**

Aircraft added successfully

**Actual Output:**



**Test Case: Get aircraft details.**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

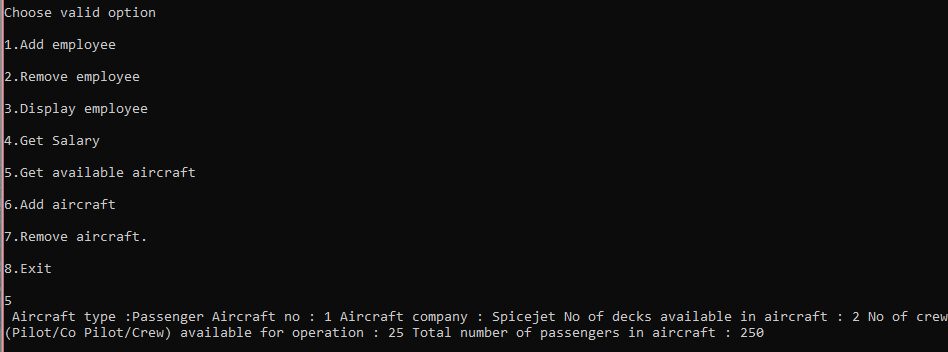
8.Exit

5

**Expected Output:**

Aircraft type: Passenger Aircraft no : 1 Aircraft company : Spicejet No of decks available in aircraft : 2 No of crew(Pilot/Co Pilot/Crew) available for operation : 25 Total number of passengers in aircraft : 250

**Actual Output:**



**Test case: Remove Aircraft**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

7

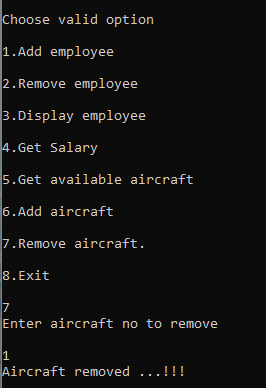
Enter aircraft no to remove

1

**Expected Output:**

Aircraft removed ...!!!

**Actual Output:**



**Test Case: Exit the Portal**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

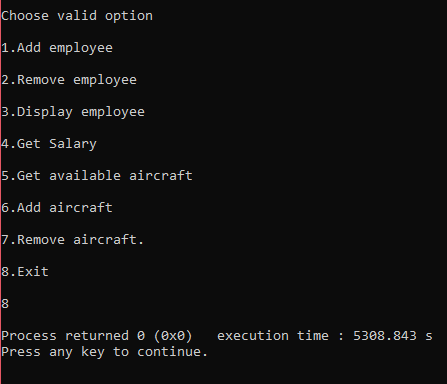
8

**Expected Output:**

Process returned 0 (0x0) execution time : 5308.843 s

Press any key to continue.

**Actual Output:**



**EXCEPTION HANDLING USED IN THE PROJECT**

**In Reservation:**

**Test Case: Reservation system**

**Input:**

Choose option below

1.Make booking

2.Review booking

3.Cancel booking

4.Exit

5

**Expected Output:**

Choose valid option

**Actual Output:**



**Test Case : Making the booking**

**Input:**

Choose source for boarding flight

1.Delhi(DEL)

2.Mumbai(BOM)

3.Bangalore (BLR)

4.Patna(PAT)

5.Kolkata(CCU)

6.Hyderabad(HYD)

7.Chennai(MAA)

8.Goa

9

**Expected output:**

Choose valid option

**Actual Output:**



**Test Case: Reviewing a Booking**

**Input:**

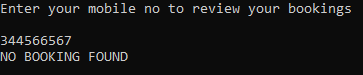
Enter your mobile no to review your bookings

344566567

**Expected output**

NO BOOKING FOUND

**Actual Output:**



**Test Case: Cancelling the Booking**

**Input:**

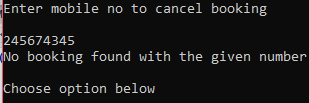
Enter mobile no to cancel booking

245674345

**Expected output**

No booking found with given number

**Output:**



**In Management Section:**

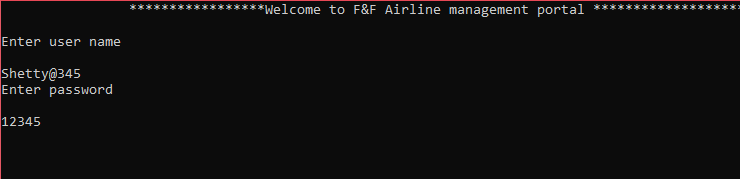
**Input: (Incorrect)**

Enter user name

Shetty@345

Enter password

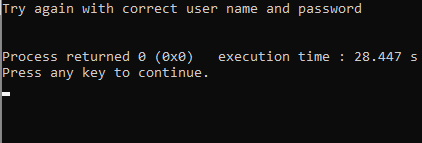
12345



**Expected Output:**

Try again with correct user name and password.

**Actual Output:**



**Test case: Removing Employee**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

2

Enter employee id

4566

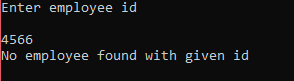
**Expected Output (When employee NOT found):**

Enter employee id

4566

No employee found with given id

**Actual Output (When employee NOT found):**



**Test Case: Getting Salary Information**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

4

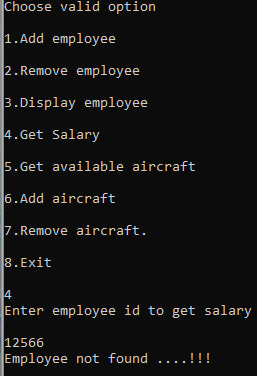
**Expected Output: (When there is no data)**

Enter employee id to get salary

12566

Employee not found ...!!!

**Actual Output:**



**Test case: Remove Aircraft**

**Input:**

Choose valid option

1.Add employee

2.Remove employee

3.Display employee

4.Get Salary

5.Get available aircraft

6.Add aircraft

7.Remove aircraft.

8.Exit

7

Enter aircraft no to remove

2

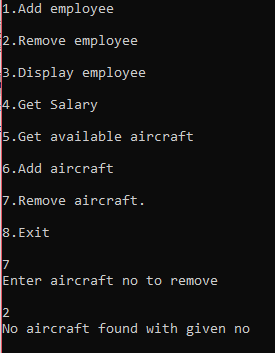
**Expected Output (When data not present)**

Enter aircraft no to remove

2

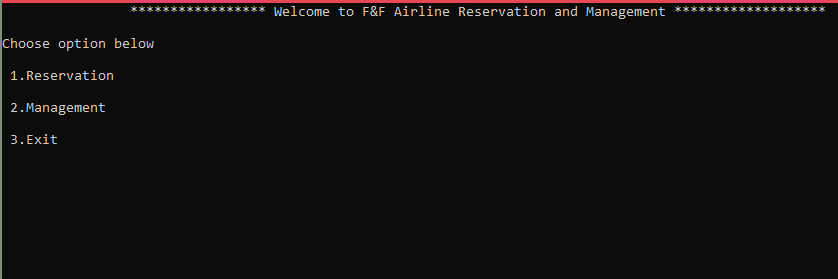
No aircraft found with given no

**Actual Output**

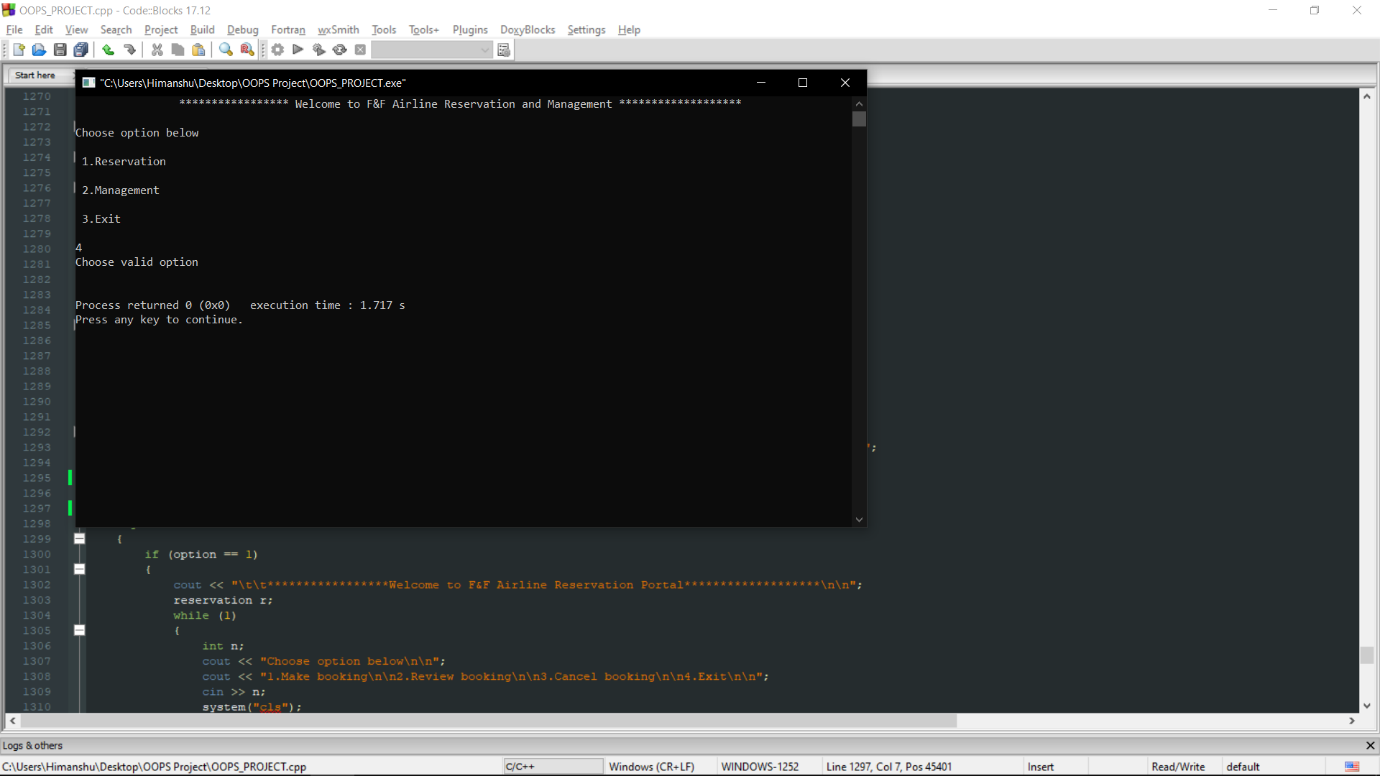


1. **Implementation**

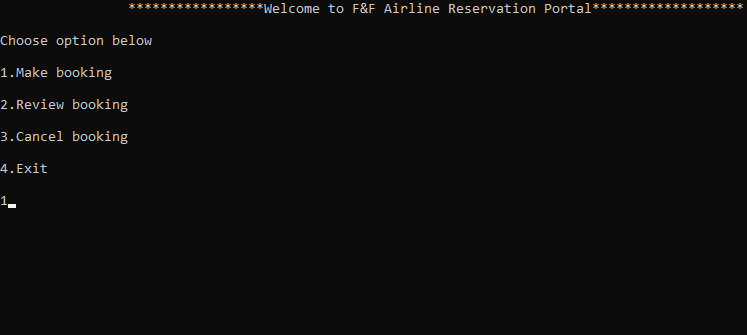
**1. Main**



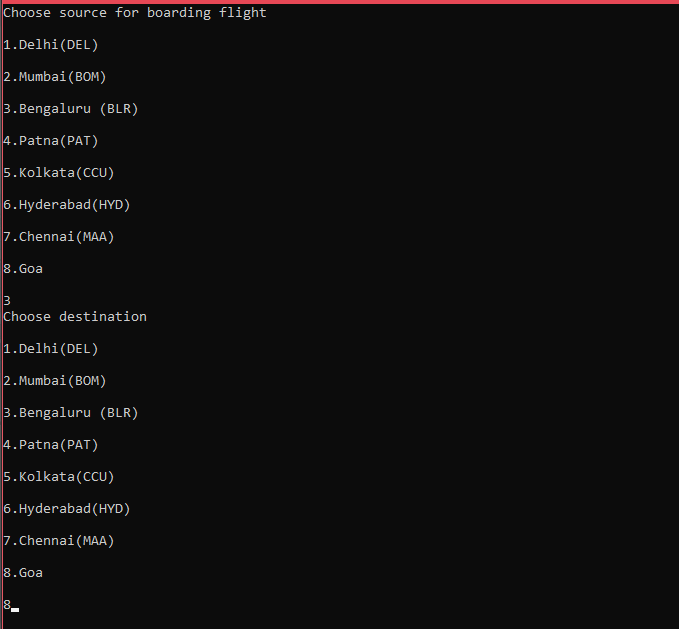
* 1. **Exception handling in main**



**2.Reservation options**



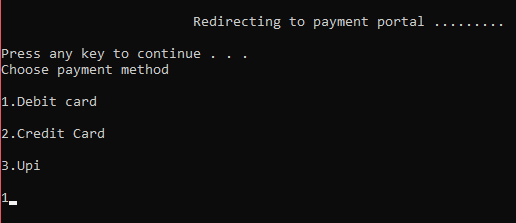
**2.1 Make booking instance to choose source and destination**



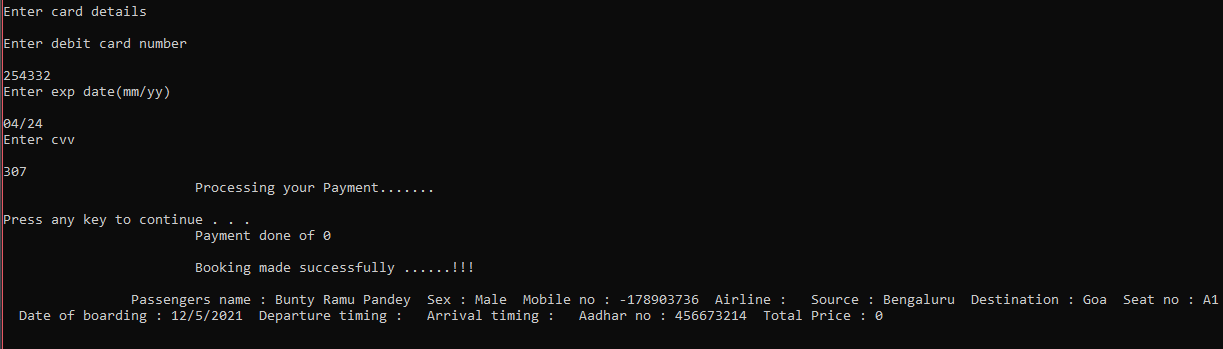
**2.1.1 In selection of source and destination exception found**



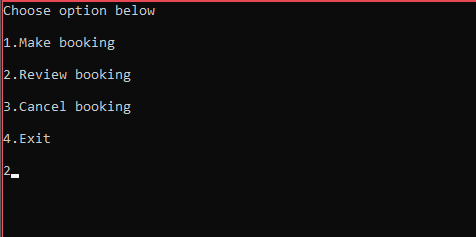
**2.2 Payment portal**



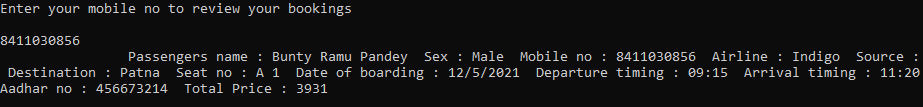
**2.3 Payment processed**



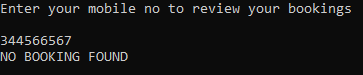
**3.Review instance**



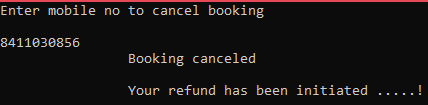
**3.1 Check for your desired booking (Found)**



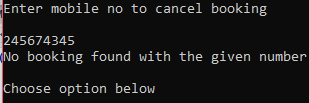
* 1. **Check for your desired booking (Not Found)**

****

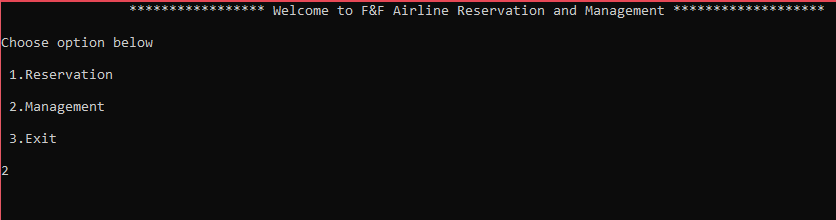
**4.Cancel instance (Booking Found)**

****

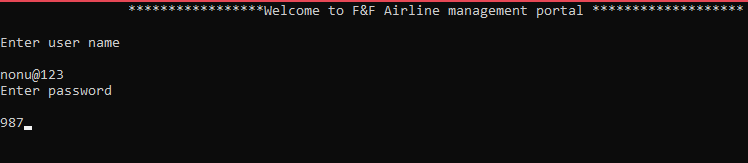
**4.1 Cancel instance (Booking Not Found)**

****

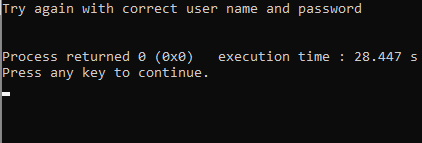
**5.Mangement**

****

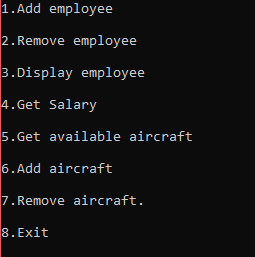
**6.Credential’s authorization**

****

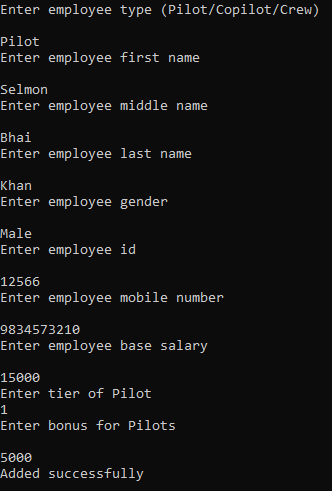
**6.1 Credential’s authorization invalid**

****

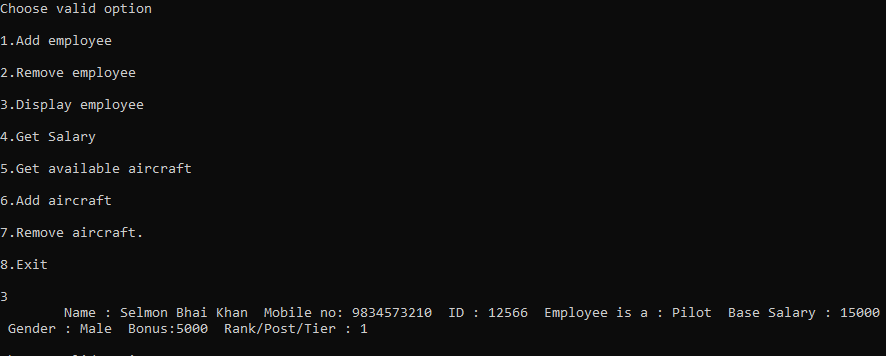
**7.Mangement object created (Credential’s authorized)**

****

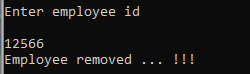
**7.1 Add employee**

****

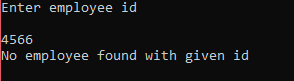
**7.2 Display employees**

****

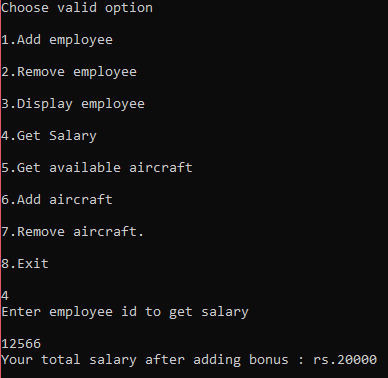
**7.3 Remove employee**

****

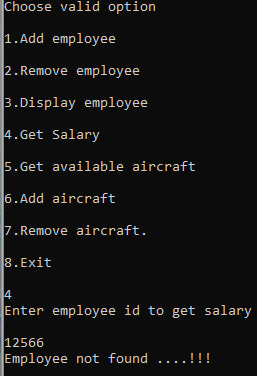
**7.3.1 Remove employee exception (Employee not found)**

****

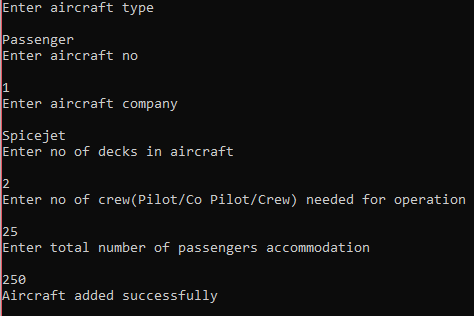
**7.4 Get total salary of the given employee**

****

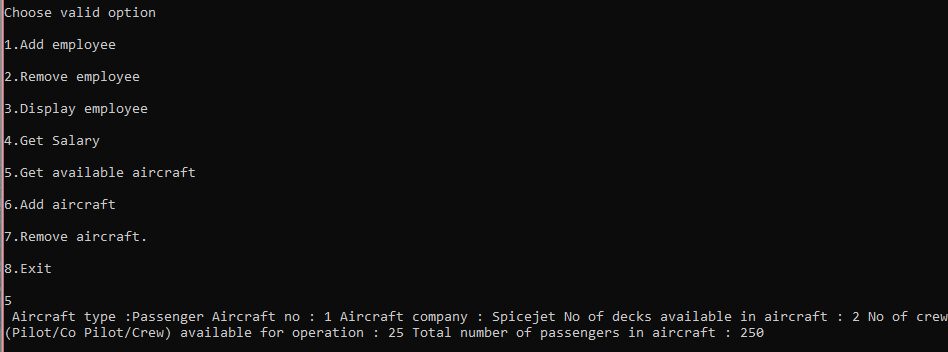
**7.5 Get total salary of the given employee (Exception employee not found)**

****

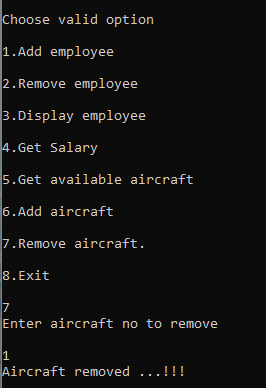
**8.Aircraft carrier management**

****

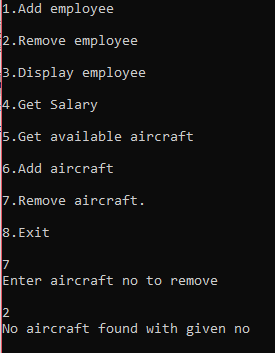
**8.1Available aircrafts**

****

**8.2 Discard/Remove aircraft (Aircraft found)**

****

**8.3 Discard/Remove aircraft (Aircraft Not found)**

****