

DELHI TECHNOLOGICAL UNIVERSITY



OOSE PROJECT REPORT

AIRLINE RESERVATION SYSTEM

SUBMITTED TO:
Dr. ROHIT BENIWAL

Saarthak Jain, Samar Saini
2K18/CO/306, 2K18/CO/312

TABLE OF CONTENT

S. No	Experiment Name	Page number
1	Problem statement of the case study.	2
2	Elicitation techniques for different domains.	3
3	Initial requirement document of the case study.	5
4	Software Requirement Specification document of the case study.	8
5	Use case diagram of the case study.	16
6	Use case description of the case study.	17
7	Class diagram of the given case study.	21
8	Sequence diagram of the given case study.	22
9	Collaboration diagram of the given case study.	26
10	Activity diagram of the given case study.	33
11	Test cases/ Test report of the given case study.	34

PROBLEM STATEMENT:

- The purpose of this document is to build an online system to manage flights and passengers to ease the flight management.
- Users of the system should be able to retrieve flight information between two given cities with the given date/time of travel from the database. The system will support two types of user privileges, Customer, and Employee.
- Customers will have access to customer functions, and the employees will have access to both customer and flight management functions. The customer should be able to make a flight reservation for either one-way trip or a round trip on a particular date. He will be asked for the confirmation before making his reservation. Customer will also be able to cancel the existing reservation and can also view the details of already reserved flights.
- On the other hand, Employee can access details of all the customers as well as of the flights, view flight schedule and calculate sales for a given flight. The administrator access will allow an employee to add, delete and update the information. He will be able to add/delete a flight, update fare for flights and update arrival/departure time of flights. Each flight has a limited number of available seats. There are a number of flights which depart from or arrive at different cities on different dates and time.

PROCESS MODEL:

For the development process of this system, we have chosen the *Waterfall model*.

We decided to go with this model because:

- Requirements are very well documented, clear and fixed at the start of the software making.
- There are no ambiguous requirements.
- Technology used in making this software is not new and well understood by the developers.
- Since it is not a real project to be used in the market, we can use Waterfall model

REQUIREMENT ELICITATION:

1. SURVEY:

The survey was carried out among the stakeholders and was conducted to gather information regarding their preference in various online airline reservation systems and how often they consider the online booking system.

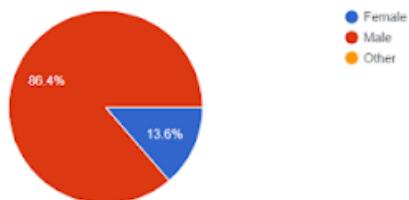
Survey Link:

[https://docs.google.com/forms/Airline Reservation System](https://docs.google.com/forms/Airline%20Reservation%20System)

Observations and Result:

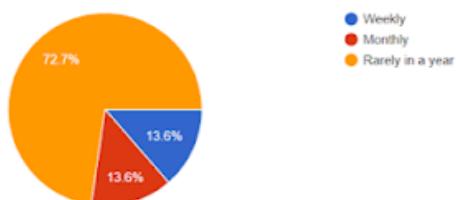
Please select your gender

22 responses



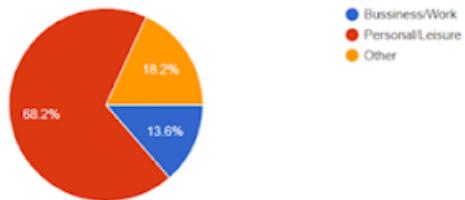
How often do you use flights to travel?

22 responses



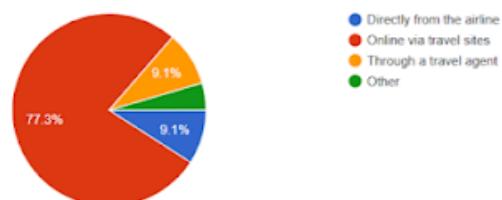
For what reason do you typically fly

22 responses



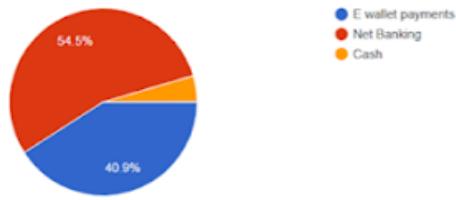
How do you typically purchase your airline tickets

22 responses



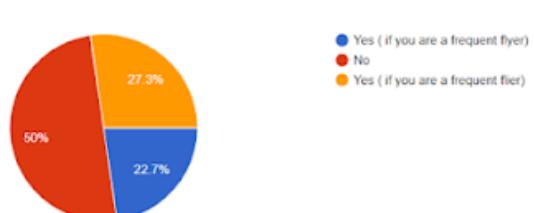
How do you prefer to pay for airline tickets?

22 responses



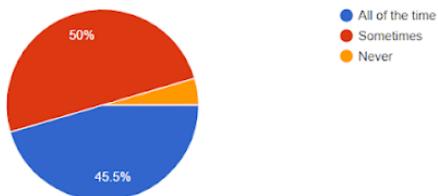
Are you comfortable sharing your location for the purpose of future suggestions and cheaper price notifications?

22 responses



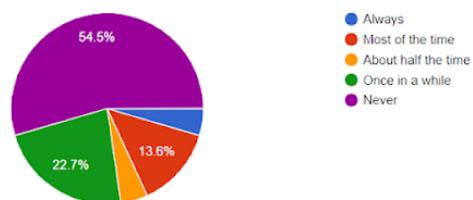
How often do you check for the pricing of tickets on different sites to get the cheapest price?

22 responses



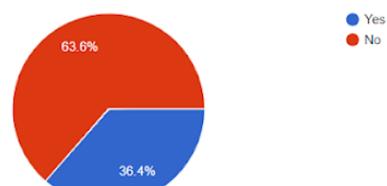
How often do you pay additional fee to choose your own seat on the plane

22 responses



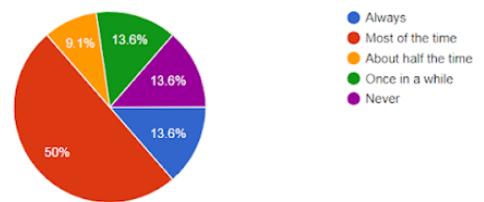
Have you ever faced refund problems against your ticket being canceled?

22 responses



How often do you pack less than you would like in order to avoid baggage fees while travelling

22 responses



Approximately 77% of people prefer online reservations. But due to some trust issues, people use other methods for booking. The major concern for an online airline reservation system is secure booking, smooth cancellations, and genuine price for traveling. People taking part in the survey prefer net banking for payment, mostly people use flight once in a year for personal reasons. People generally don't pay extra charges for seat preference as well as for extra luggage.

Initial Requirement Document

Introduction:

Airline Reservation Systems (ARS) used to be standalone systems. Each airline had its own system, disconnected from other airlines or ticket agents, and usable only by a designated number of airline employees. Travel agents in the 1970s pushed for access to the airlines' systems. Today, air travel information is linked, stored, and retrieved by a network of Computer Reservations Systems (CRS), accessible by multiple airlines and travel agents.

Goals:

This proposal is for performing a case study to be undertaken to design and implement an airline reservation system. Reviewing the literature of reservation systems and exploring the advantages and limitations of reservation systems in real-life situations will also be a part of the case study.

Specifications:

Interviews and questionnaire methods will be a major part of the data collection phase, these data collection methods will help to better understand the existing system in use. Case tools and data flow diagrams are intended to be used during the development process to simulate the process of airline reservation and ticket booking. The case study will also contain ER diagrams and SRS for the system.

Milestones:

The current system is manual, this system is slow, time-consuming and it is very difficult for each person to book through office agents. Users inquire about the tickets through phones, and it is tough for the user to remember all the details that they received through phones. It is very difficult to calculate how many people registered and how many seats on a particular plane are vacant. This requires quite a lot of time and wastage of money as it requires quite a lot of manpower to do. The general objective is to automate the process of airline ticket reservation, booking, and airline management hence minimize errors resulting from manual system operations.

Scope of Study:

Minimize repetitive work done by the system administrator and reservation clerks. Maintain consistency among different access modes, e.g. by phone, by web, at the information desk, and across different physical locations. In addition, the outcome of this study will provide a basis for developing the appropriate approach to the problems associated with air traveling operations in relation to Airline Reservation Systems.

Specific Objectives:

- To study the current system identifying its inefficiencies.
- To design an online airline reservation information system to facilitate online booking and flight schedules.
- To determine both functional and non-functional requirements for the new system.
- To provide a user-friendly interface.
- To implement the developed web-based airline information system.
- To test and validate the developed system by use of a case study.
- To provide flexibility and more security in the payment transaction.
- To provide some additional features as per requirements.

- The system should minimize repetitive work done by the system administrator and reservation clerks.
- The system should maintain customer information in case of emergency, e.g., flight cancellation due to inclement weather.

Functional Requirements:

1. User registration:

This case describes the scenario where the user registers with the application by providing all the necessary details in order to make reservations, motels, special packages.

2. User login:

In this user logs into the application with the username and password he has provided during registration.

3. Contact the company:

Users can contact the company for any info.

4. Booking Instructions:

Users can view the instructions for booking flights.

5. Booking packages:

Users can either book a two-way trip or a one way trip at the time of reservation.

6. Login/Logout:

Administration of the application logs into the system and logout after the work is done.

7. Add/Delete/modify customer info:

Admin adds, deletes, or modifies customer info in the system database.

8. Cancellation of reservation:

Admin handles the cancellation of reservations by the customers.

9. Email confirmation:

Admin sends an email confirmation to the customers of the application.

Non-functional requirements:

Performance Requirements

Queries filed by users should be responded quickly by the ARS. When a user looks for a flight departing from one airport to another airport, the ARS should quickly return the results. As the ARS application is not that big, it should show six-seven results at a time on each page to the user, when he/she searches for some data. Time taken by the application to answer the requests of users should be less than two and a half seconds.

Safety Requirements

If a large chunk of the database suffers significant harm due to accidental failure, for example a storage device crash, a recovery technique should be present that recovers a former copy of the database that was backed up to backed-up storage and recreates a more present state by reapplying or repeating the operations of committed transactions from the backed-up log, up until the moment of failure.

Security Requirements

Security systems, like so many other systems, require storage in the form of database. But, due to the unique needs of the security sector, suppliers must hand pick their database partner.

Software Quality Attributes

- **AVAILABILITY of system:** Users should be able to access application on any time of the day for booking or cancelling the tickets.
- **CORRECTNESS of system:** The flights available on application should depart from the correct terminal and also arrive at the proper destination.
- **MAINTAINABILITY of system:** Correct schedules of flights should be maintained by the administrators and flight management crew.
- **USABILITY of system:** Flight itineraries should be designed to meet the requirements of as many consumers as possible.

SYSTEM REQUIREMENTS SPECIFICATIONS

FOR THE

<AIRLINE RESERVATION SYSTEM>

By:

Samar Saini (2K18/CO/312)

Saarthak Jain (2K18/CO/306)

TABLE OF CONTENTS

- 1) Introduction
 - 1.1) Purpose
 - 1.2) Document Conventions
 - 1.3) Intended Audience and Reading Suggestions
 - 1.4) Project Scope
 - 1.5) References
 - 1.6) Overview
- 2) Overall Description
 - 2.1) Product Perspective
 - 2.2) Product Features
 - 2.3) User Characteristics
 - 2.4) Operating Environment
 - 2.5) Constraints
 - 2.6) Assumptions and Dependencies
- 3) External Interface Requirements
 - 3.1) User Interfaces
 - 3.2) Hardware Interfaces
 - 3.3) Software Interfaces
- 4) System Features
 - 4.1) Functions Requirements
- 5) Non-Functional Requirements
 - 5.1) Performance Requirements
 - 5.2) Safety Requirements
 - 5.3) Security Requirements
 - 5.4) Software Quality Attributes

1) INTRODUCTION

1.1 PURPOSE

The goal of this article is to create an online platform for handling flights as well as travellers in order to make flight management easier.

1.2 DOCUMENT CONVENTIONS

The mentioned conventions are used in this document:

- ARS: Airline Reservation System
- ERD: Entity Relationship Diagram
- DBMS: Database Management System

1.3 INTENDED AUDIENCE AND READING SUGGESTIONS

This whole project is a model for a flight management system that is confined to the university premises. This is expected to be beneficial for both the flight managing crew as well as the passengers.

1.4 PROJECT SCOPE

The goal of said online ARS is to make flight management much easier and also to build a comfortable and user-friendly interface for customers attempting to purchase air tickets. Including its flight management and booking functionalities, the platform is constructed on a relational database. We will have a database system that would accommodate hundreds of major locations worldwide along with thousands of flights from numerous airlines. And thus, we want to ensure a delightful customer experience as well as an efficient and fast system for the management.

1.5 REFERENCES

Project Report: Contains the different views of the system using Sequence Diagrams, Activity Diagrams etc. and the details of the requirement elicitation for the development.

1.6 OVERVIEW

Section 2 of this document describes overview of system in terms of general characteristics of the system, information about possible users of the system, possible constraints on the system, operating environment of the system etc. The Section 3 describes in detail, the different interfaces and their requirements. The Section 4 describes the system features and their relationships in brief. Finally, the Section 5 discusses about the non-functional requirements like safety, performance, etc requirements.

2) OVERALL DESCRIPTION

2.1 PRODUCT PERSPECTIVE

The following records will be maintained within a distributed airline DBMS:

- **Flight Details**

This comprises the source and destination flight terminals, the schedule, total number of seats reserved or vacant for each class, and so on.

- **Customer/Passenger Details**

This comprises the information like customer id, name, address, phone number and so on which may be used to retain the customer's records in case of an emergency or for any other purpose.

- **Reservation Details**

This contains the passenger information, a PNR numbers, flight numbers, the date of reservation, the date of travel, billing information, and so forth.

- **Employee Details**

This contains the details of the flight management crew like name, ids, phone numbers, etc.

2.2 PRODUCT FEATURES

The ARS supports following functions:

- Functions by which customers can book or cancel an online ticket for a particular date, place, class and check the ticket status.
- Functions by which employees get the status of all customers and flights.
- Functions by which employee can add/delete a flight.
- Functions by which employee can update a flight's information/status like updating the departure/arrival time of a flight, etc.

The access to these different functions by different users is restricted.

2.3 USER CHARACTERISTICS

Users of the system will be permitted to get flight info connecting two specified locations on the specified date and time of travel. Customer and Employee user privileges will be supported by the platform. Customers will be able to use customer services, while staff will be able to access both customer and flight management capabilities

The Customer should be able to do the following functions:

- Make a new reservation based on availability
 - Flexible Date/time and other requirements
 - Payment and Confirmation
- Cancel an existing reservation
- View his itinerary

The Employee should have following management functionalities:

- Customer Functions:
 - Get all the passengers and their details, who have seats reserved on a given flight.
 - Get all flights and their details for a given airport and time.
- Administrative Functions:
 - Add and Delete a flight
 - Update fare for flights.
 - Update departure/arrival.

As a result, each trip will have a fixed number of available seats in various classes, and there are a number of flights that depart or arrive at various places on various days and times.

2.4 OPERATING ENVIRONMENT

Operating environment for the airline management system is as listed below.

- Distributed database
- Operating system: Windows.
- Database: SQL+

- Platform: To be decided

2.5 CONSTRAINTS

User is required to remember the login ID and password for his account. If lost, user will not be able to access his account again. Customers will be allowed to check the ticket status and book ticket only through internet. Payment for the tickets can only be made through net banking.

2.6 ASSUMPTION AND DEPENDENCIES

As of now, there are no assumptions. This article will be updated in future editions.

3) EXTERNAL INTERFACE REQUIREMENTS

3.1 USER INTERFACES

The ARS project has two sorts of users. The first is the customer, while the second is the employee/administrator. Both Customer and Employee user interfaces would be graphical. The following software is being used for functionalities:

- Front-end software is not yet decided
- Back-end software we are using is SQL+

3.2 HARDWARE INTERFACES

The machines should be running Windows and also have a browser that supports CGI, HTML, and JavaScript. For interactivity, the computer must communicate with a conventional output device, a keyboard, and a mouse.

3.3 SOFTWARE INTERFACES

The programme should be compatible with the Windows operating system. SQL+ is required because the programme requires a database to hold all of the client information, airline information, and reservation information.

The decision for the coding language to implement the coding of the software is yet to be made. It will be either C++/Python/PHP.

Accordingly, we will decide the IDE to work on.

4) SYSTEM FEATURES

4.1 FUNCTIONAL REQUIREMENTS

Function Name: Login

Description: This function describes how a user logs into the system.

Basic flow: User wishes to log in to the system and has to enter valid credentials necessary to log in to the system

Alternate flow: Credentials entered by the User do not follow the format prescribed for the credentials.

Function Name: Verify Password

Description: Credentials entered are verified against the database

Basic flow: Entered credentials match that of the ones present in the database
User is logged in to the system in their respective account type.

Alternate flow: If the entered credentials are not present in the database, an error message is generated.

Function Name: Check Availability

Description: Customers can check for tickets based on their requirements

Basic flow: Customers can search for tickets based on the following parameters:

- Date and Time
- Location
- Class

Alternate flow: None

Function Name: Book Ticket and Payment

Description: Provides the main payment functionality that allows Customers to book and pay for the tickets.

Basic flow: Customer selects the available tickets and initiates the payment functionality after which verification takes place.

Alternate flow: If the payment is unable to be completed due to any reason, a transactional error message is generated.

Function Name: Cancellation

Description: Customers can initiate the Cancel functionality if they wish.

Basic flow: The Customer is given the option to confirm cancellation after which they are refunded and the system records are updated.

Alternate flow: None

Function Name: View Itinerary

Description: Customer can view their flight status, details, etc.

Basic flow: Customers can now view their flight itinerary which includes the departure and arrival airports, dates and times of the flights, flight numbers, etc.

Alternate flow: None

Function Name: View Customer/Flight Information

Description: Allows Employees to access customer and flight information from the database.

Basic flow: Employees can search for information using customer id, flight id, etc depending on their needs.

Alternate flow: None

Function Name: Add Flight

Description: Allows Employees to add new flights and their respective information

Basic flow: The employee can add a new flight along with their respective information like departure and arrival times, fares, seats, etc.

Alternate flow: None

Function Name: Update Flight Information

Description: Allows employees to update information of existing flights.

Basic flow: Employees can search for the flight and perform the following:

- Update Arrival/Departure
- Update Fares
- Delete Flight

Alternate flow: If a flight does not exist in the database, an error message is generated.

5) NON-FUNCTIONAL REQUIREMENTS

5.1 PERFORMANCE REQUIREMENTS

Queries filed by users should be responded quickly by the ARS. When a user looks for a flight departing from one airport to another airport, the ARS should quickly return the results. As the ARS application is not that big, it should show six-seven results at a time on each page to the user, when he/she searches for some data. Time taken by the application to answer the requests of users should be less than two and a half seconds.

5.2 SAFETY REQUIREMENTS

If a large chunk of the database suffers significant harm due to accidental failure, for example a storage device crash, a recovery technique should be present that recovers a former copy of the database that was backed up to backed-up storage and recreates a more present state by reapplying or repeating the operations of committed transactions from the backed-up log, up until the moment of failure.

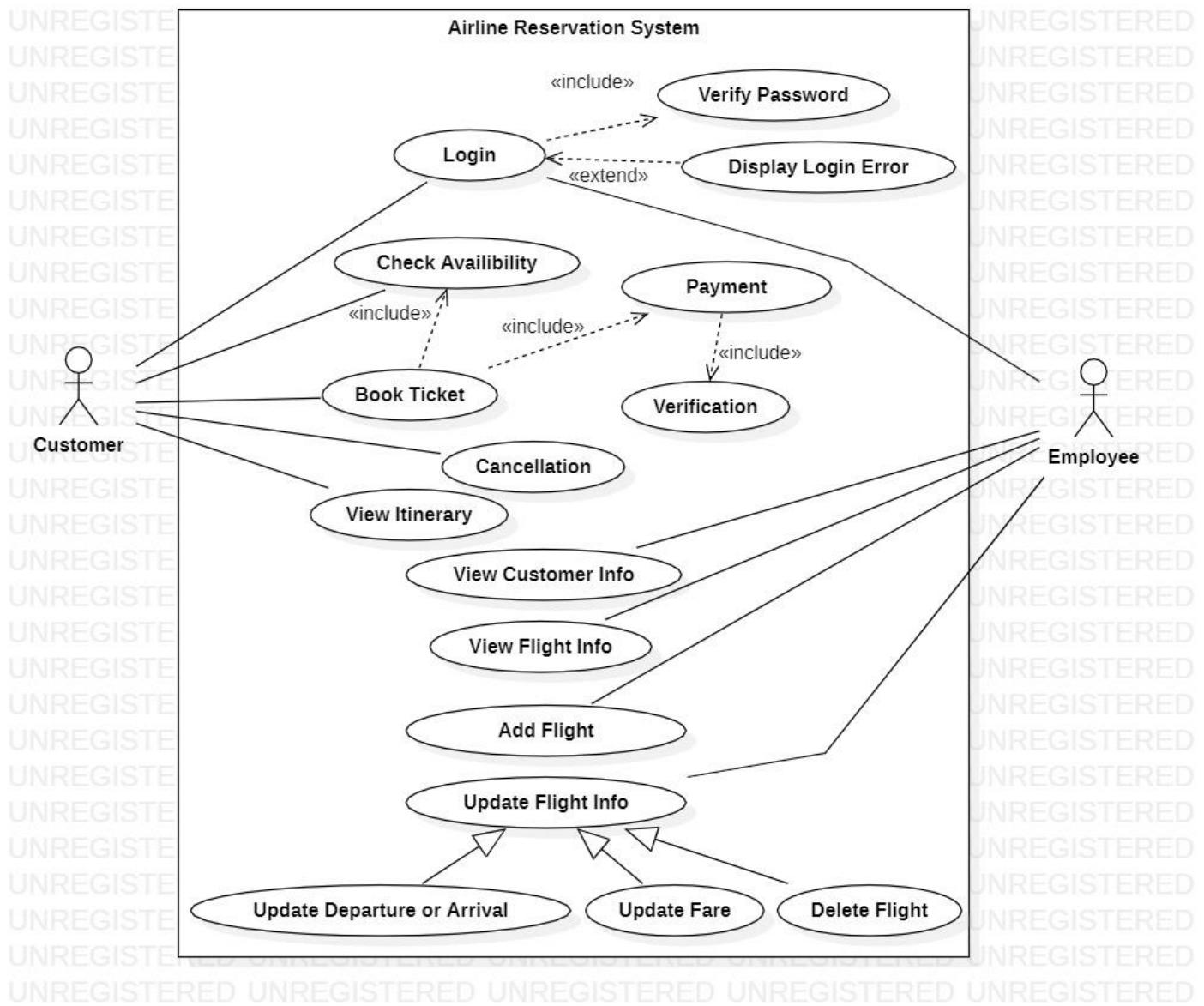
5.3 SECURITY REQUIREMENTS

Security systems, like so many other systems, require storage in the form of database. But, due to the unique needs of the security sector, suppliers must hand pick their database partner.

5.4 SOFTWARE QUALITY ATTRIBUTES

- **AVAILABILITY of system:** Users should be able to access application on any time of the day for booking or cancelling the tickets.
- **CORRECTNESS of system:** The flights available on application should depart from the correct terminal and also arrive at the proper destination.
- **MAINTAINABILITY of system:** Correct schedules of flights should be maintained by the administrators and flight-in chargers.
- **USABILITY of system:** Flight itineraries should be designed to meet the requirements of as many consumers as possible.

USE CASE DIAGRAM



USE CASE DESCRIPTION

Function Name: Login

Brief	This use case describes how a user logs into the system
Actors	Customer/Employee
Precondition	Actors should have already had an existing account of their respective type
Basic flow	Actor wishes to log in to the system and has to enter valid credentials necessary to log in to the system
Alternate flow	Credentials entered by the Actor do not follow the format prescribed for the credentials. Actor is returned to the beginning of the Use case.
Post condition	Verification of credentials

Function Name: Verify Password

Brief	Credentials entered are verified against the database
Actors	None
Precondition	Valid credentials should be present in the database
Basic flow	Entered credentials match that of the ones present in the database User is logged in to the system in their respective account type
Alternate flow	If the entered credentials are not present in the database, an error message is generated.
Post condition	Customer/Employee is allowed access to his account

Function Name: Check Availability

Brief	Customers can check for tickets based on their requirements
Actors	Customer
Precondition	The Customer must be logged in.
Basic flow	Customers can search for tickets based on the following parameters: <ul style="list-style-type: none">• Date and Time• Location• Class
Alternate flow	None.
Post condition	Customers can now book the tickets if they want.

Function Name: Book Ticket and Payment

Brief	Provides the main payment functionality that allows Customers to book and pay for the tickets.
Actors	Customer
Precondition	The Customer must be logged in and must choose the appropriate available tickets based on their need.
Basic flow	Customer selects the available tickets and initiates the payment functionality after which verification takes place.
Alternate flow	If the payment is unable to be completed due to any reason, a transactional error message is generated.
Post condition	Customer receives the booking confirmation and can log out of the system.

Function Name: Cancellation

Brief	Customers can initiate the Cancel functionality if they wish
Actors	Customer
Precondition	Customer has already booked and paid for tickets.
Basic flow	The Customer is given the option to confirm cancellation after which they are refunded and the system records are updated.
Alternate flow	None
Post condition	User receives the confirmation and can log out of the system.

Function Name: View Itinerary

Brief	Customer can view their flight status, details, etc.
Actors	Customer
Precondition	Customer must be logged in and the payment for the tickets is completed.
Basic flow	Customers can now view their flight itinerary which includes the departure and arrival airports, dates and times of the flights, flight numbers, etc.
Alternate flow	None
Post condition	None

Function Name: View Customer/Flight Information

Brief	Allows Employees to access customer and flight information from the database.
Actors	Employee
Precondition	Employee must be logged in.
Basic flow	Employees can search for information using customer id, flight id, etc depending on their needs.
Alternate flow	None
Post condition	If the entity exists, the employee can now access it.

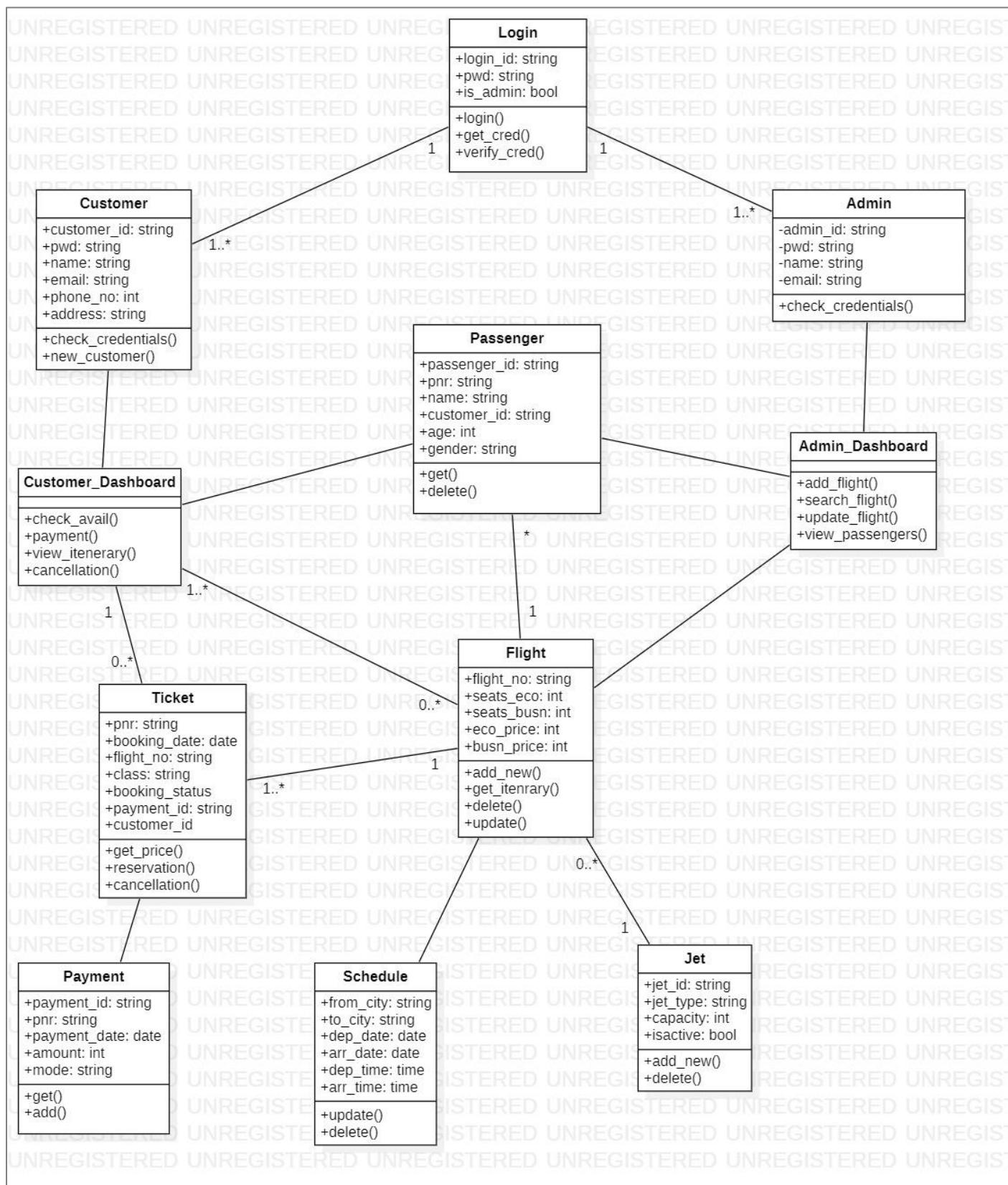
Function Name: Add Flight

Brief	Allows Employees to add new flights and their respective information.
Actors	Employee
Precondition	Employee must be logged in.
Basic flow	The employee can add a new flight along with their respective information like departure and arrival times, fares, seats, etc.
Alternate flow	None.
Post condition	Database is updated according to the changes.

Function Name: Update Flight Information

Brief	Allows employees to update information of existing flights.
Actors	Employee
Precondition	Employee must be logged in.
Basic flow	Employees can search for the flight and perform the following: <ul style="list-style-type: none">• Update Arrival/Departure• Update Fares• Delete Flight
Alternate flow	If a flight does not exist in the database, an error message is generated.
Post condition	Database is updated according to the changes.

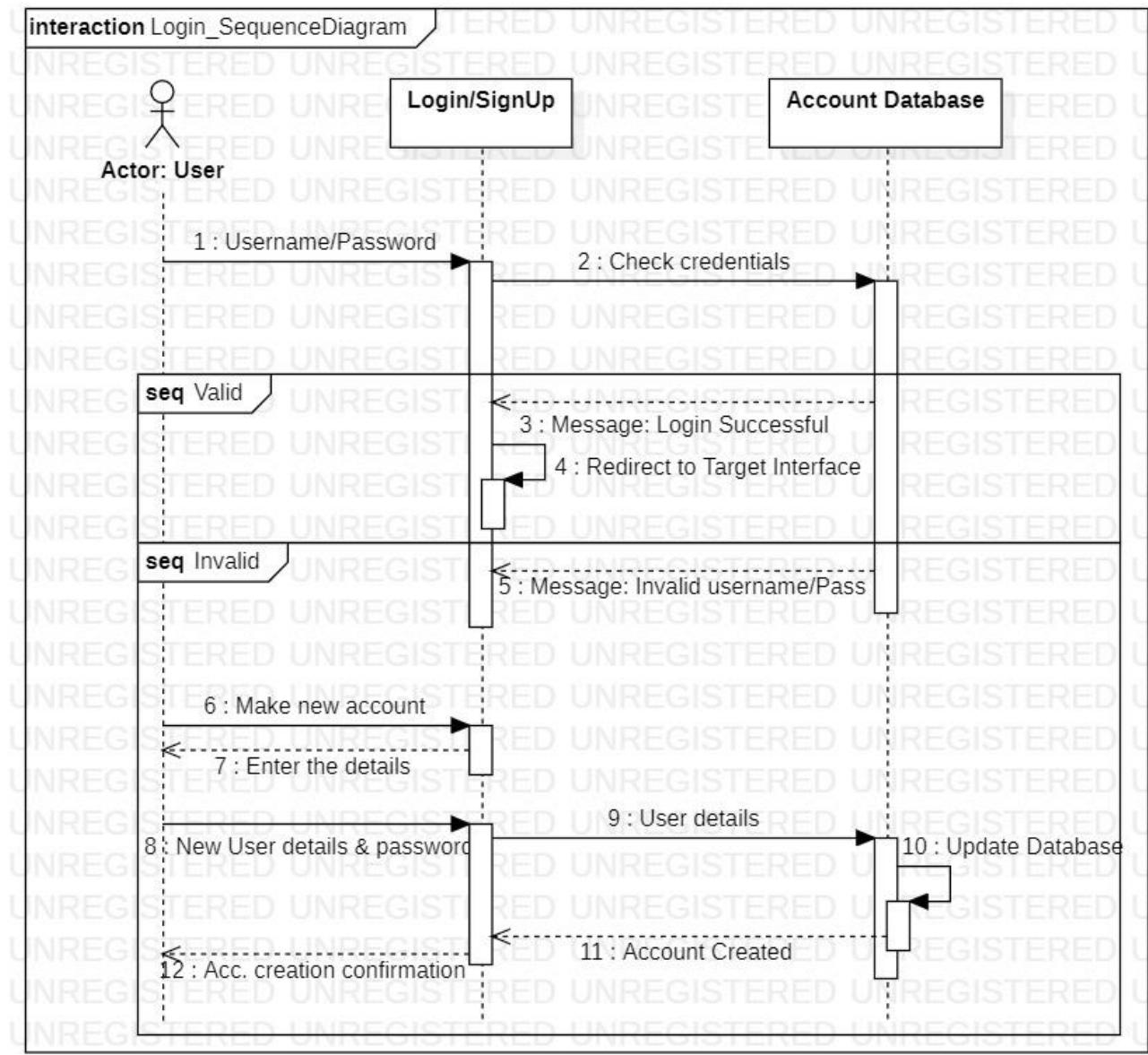
CLASS DIAGRAM



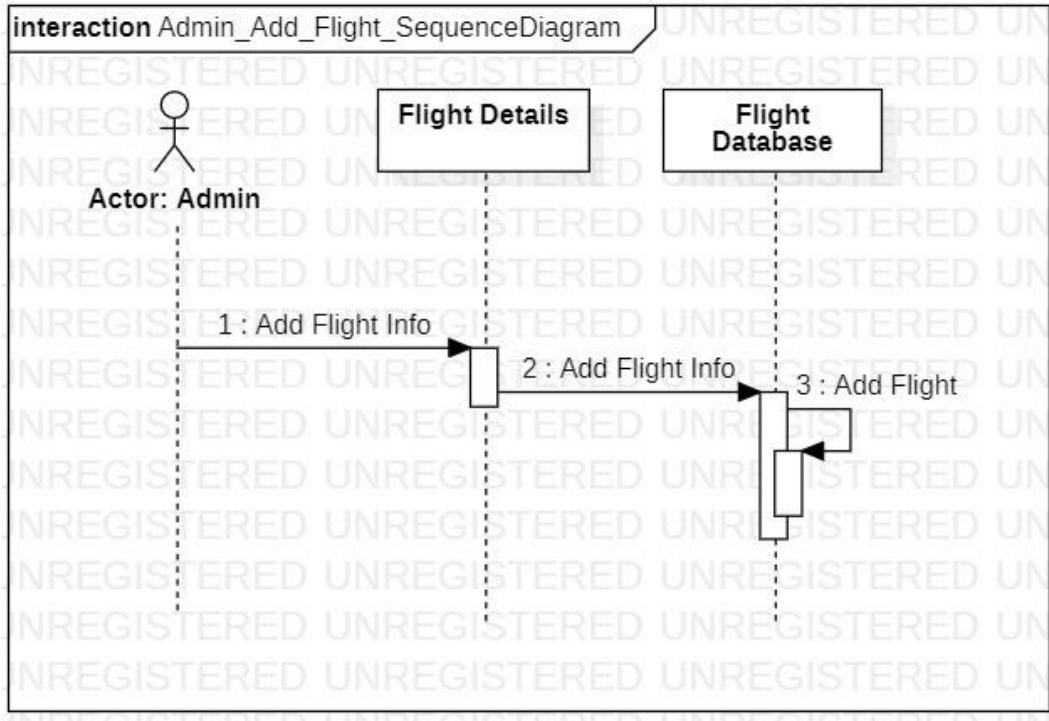
SEQUENCE DIAGRAMS

All the sequence diagrams of our project are given below:

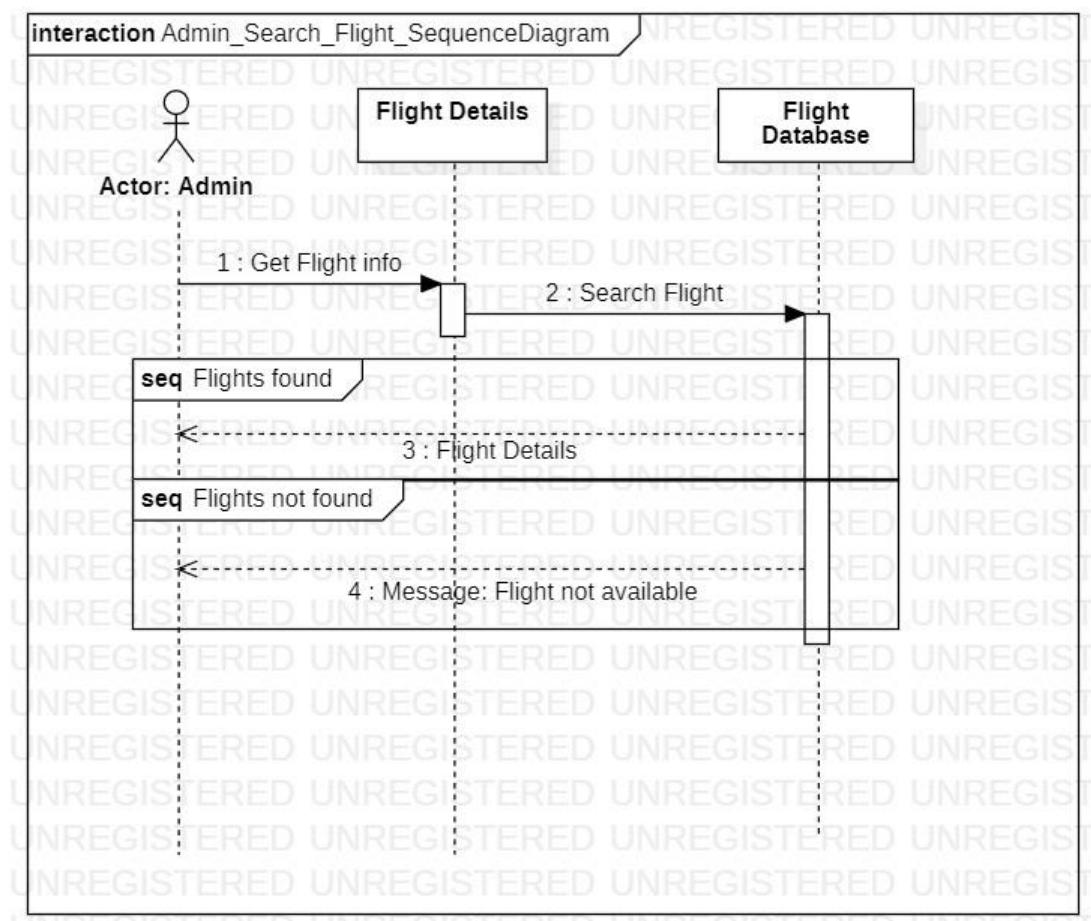
1. Login (Login and Sign Up)



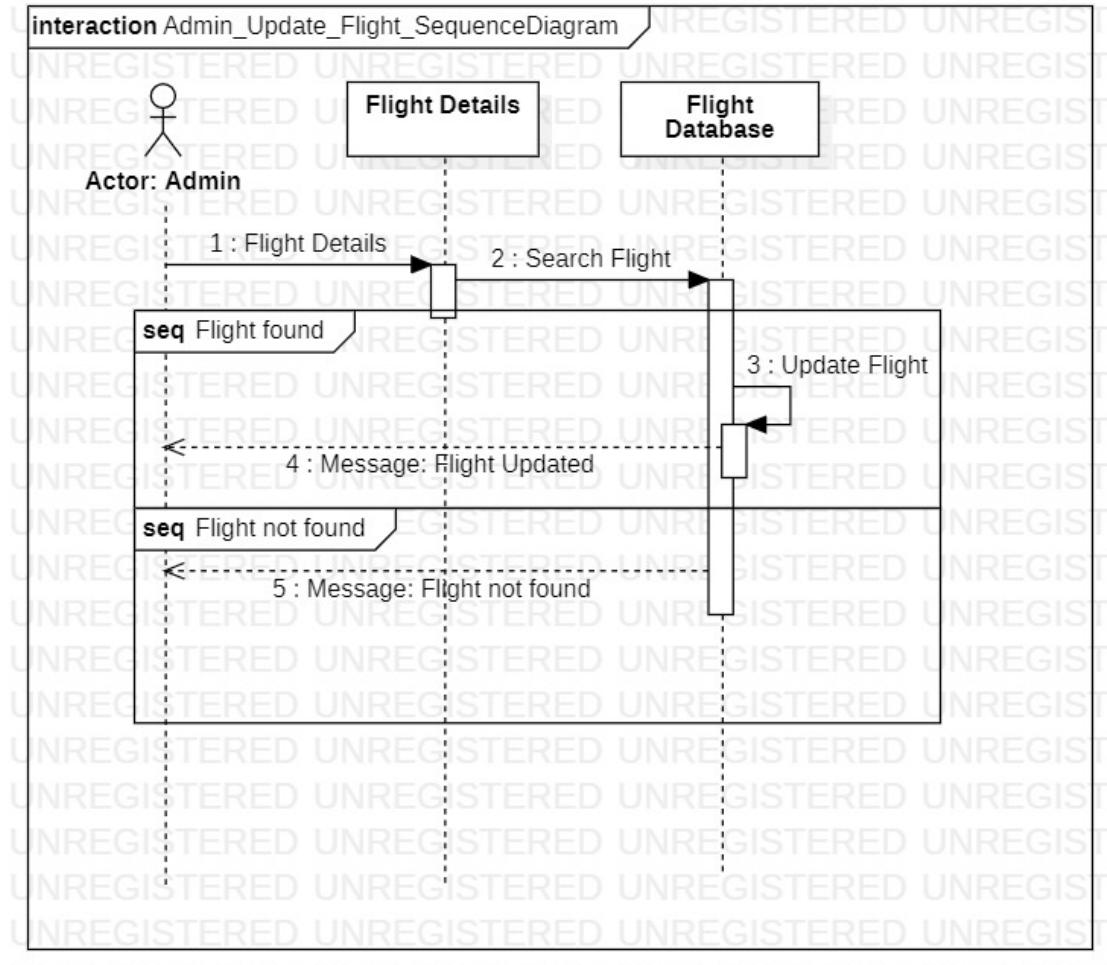
2. Admin's Add Flight Function



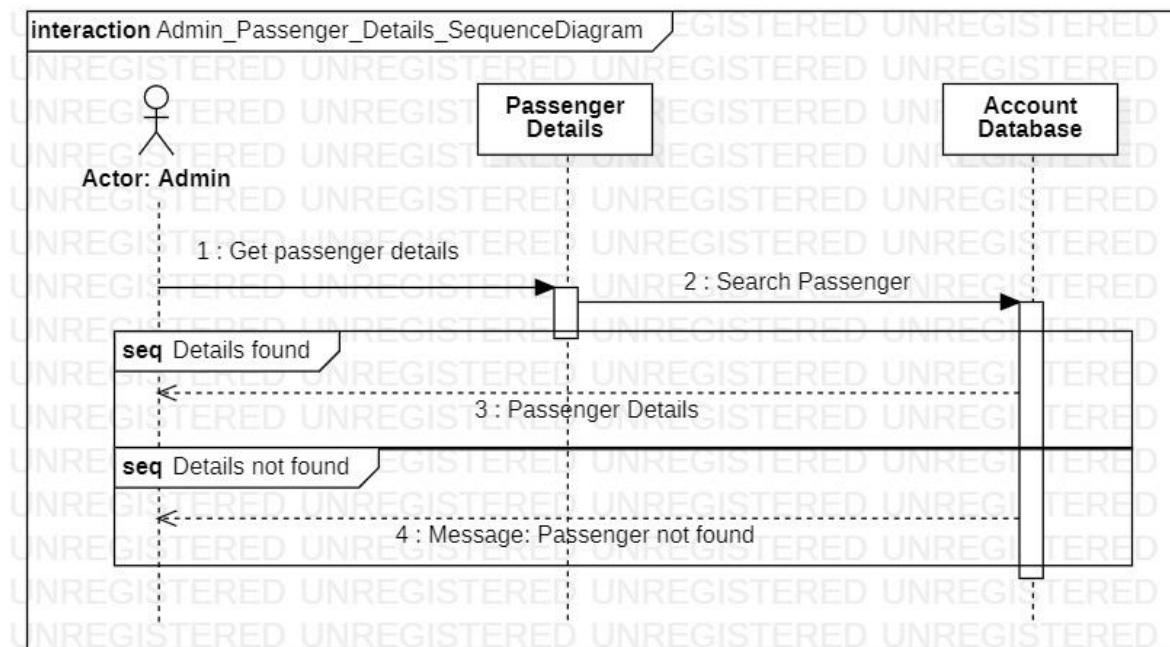
3. Admin's Search Flight Function



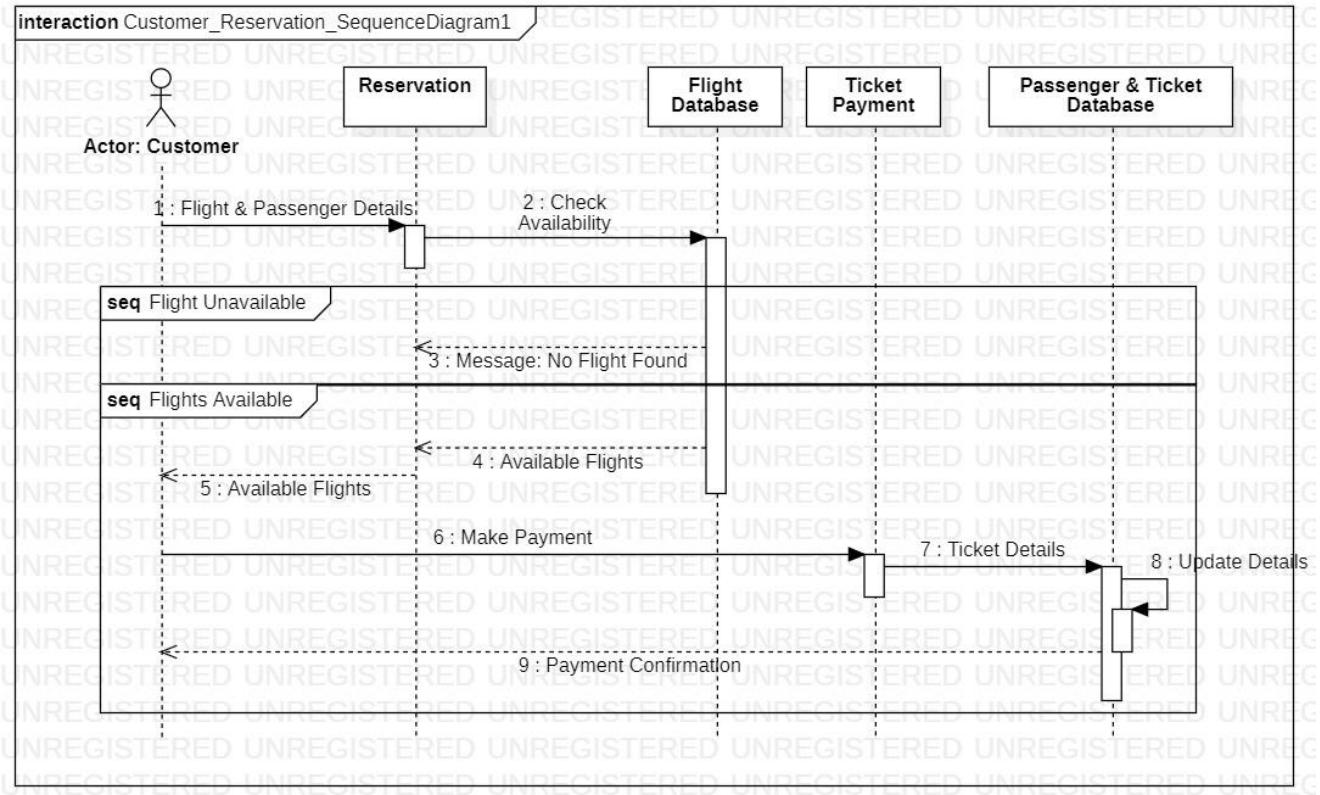
4. Admin's Update Flight Function



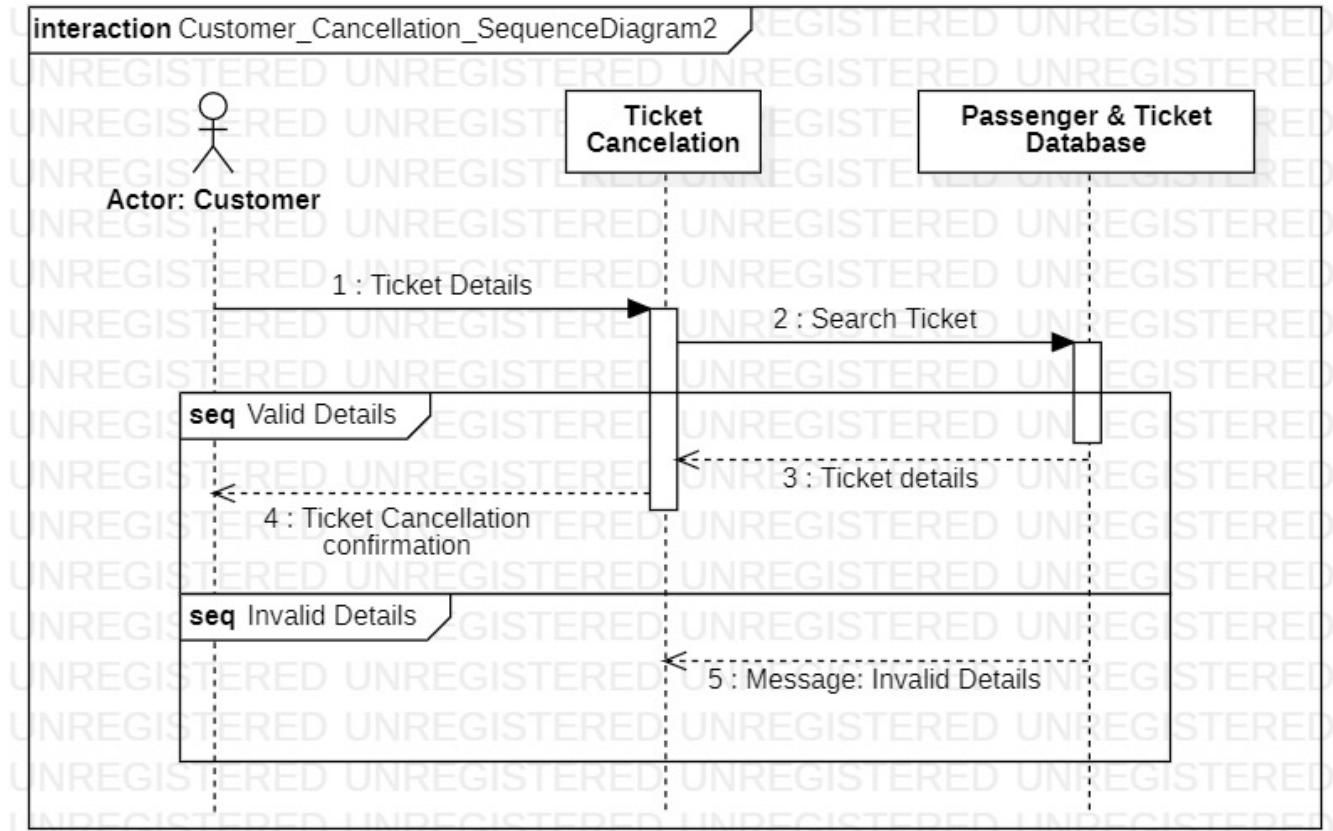
5. Admin's Get Passenger Details Function



6. Customer's Reservation Function

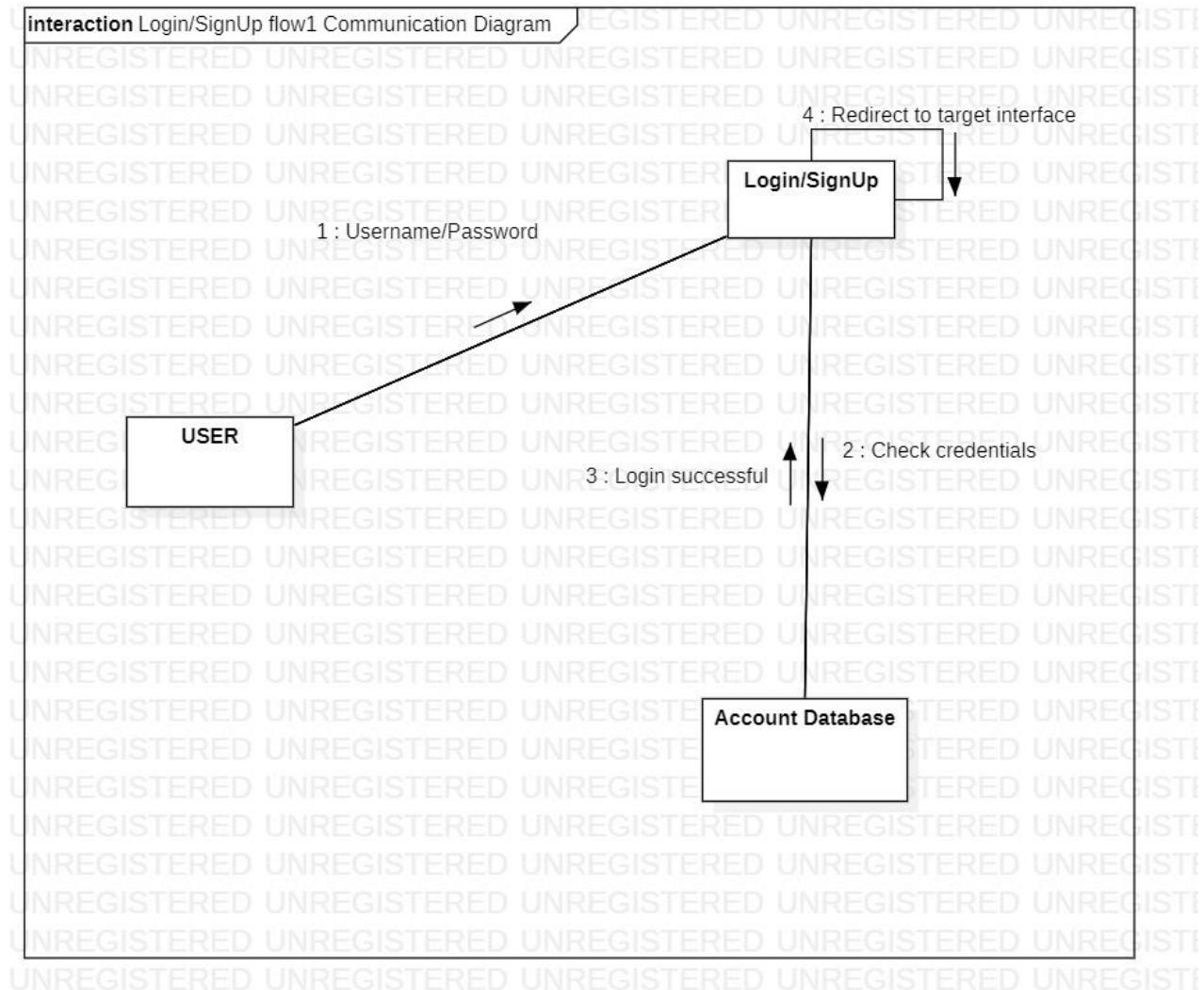


7. Customer's Ticket Cancellation Function

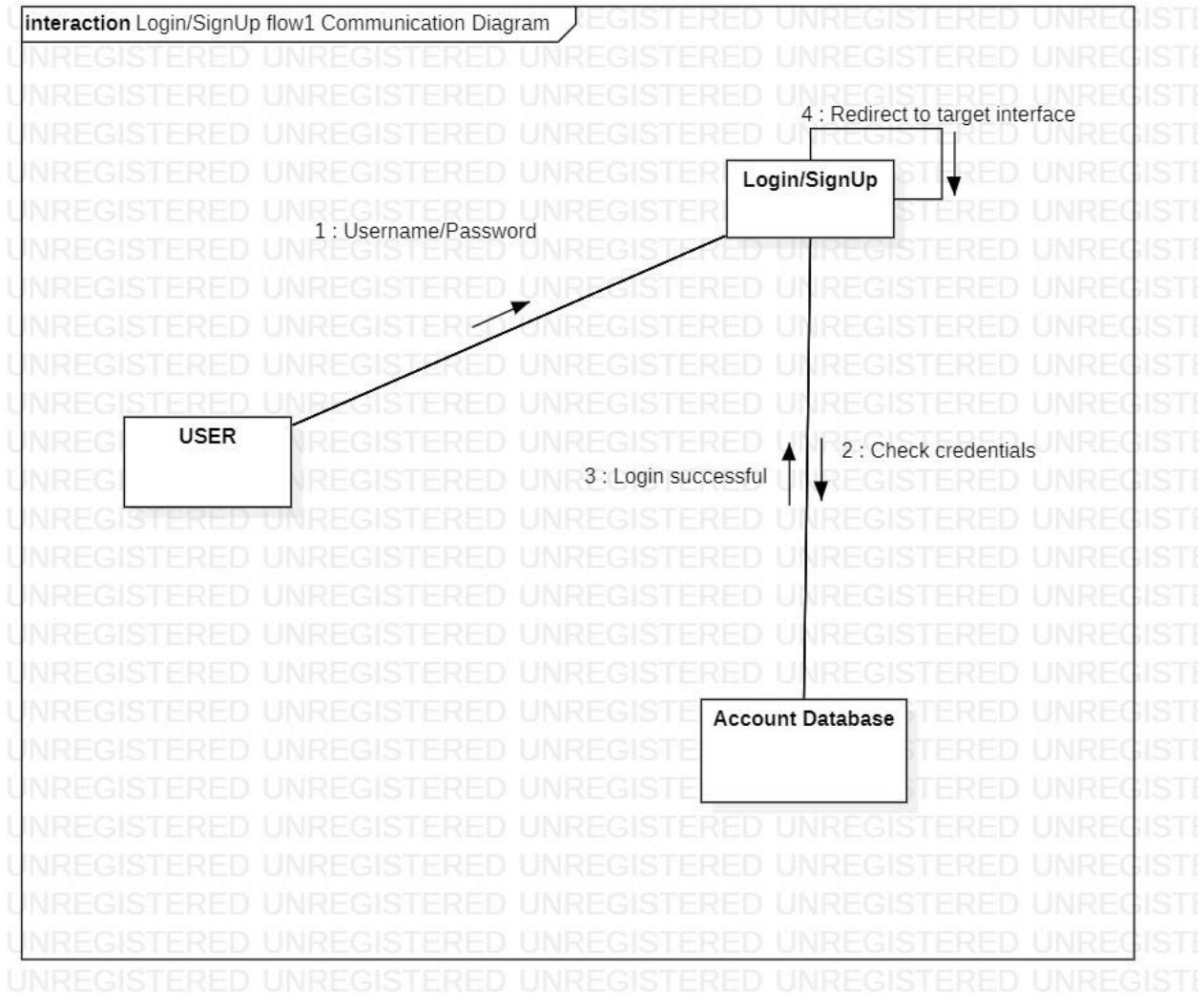


COLLABORATION DIAGRAMS

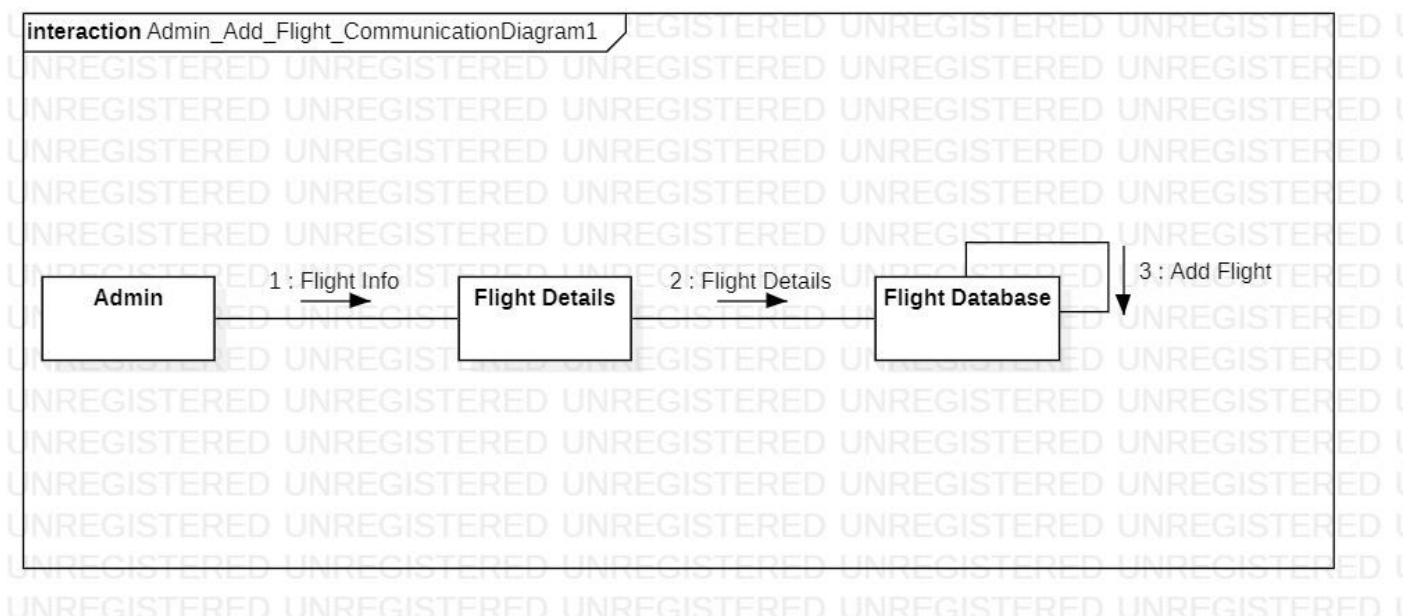
1. Login flow 1:



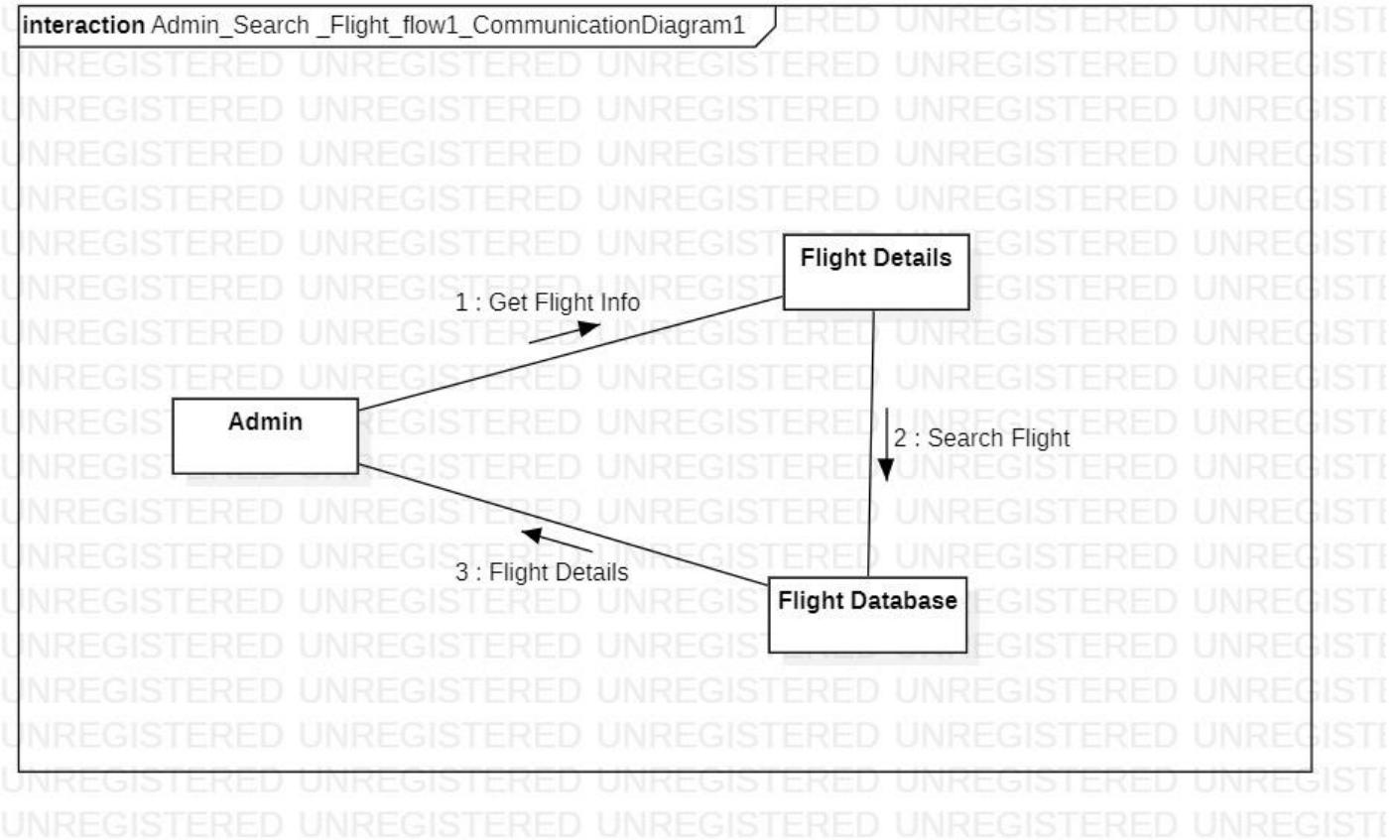
2. Login flow 2:



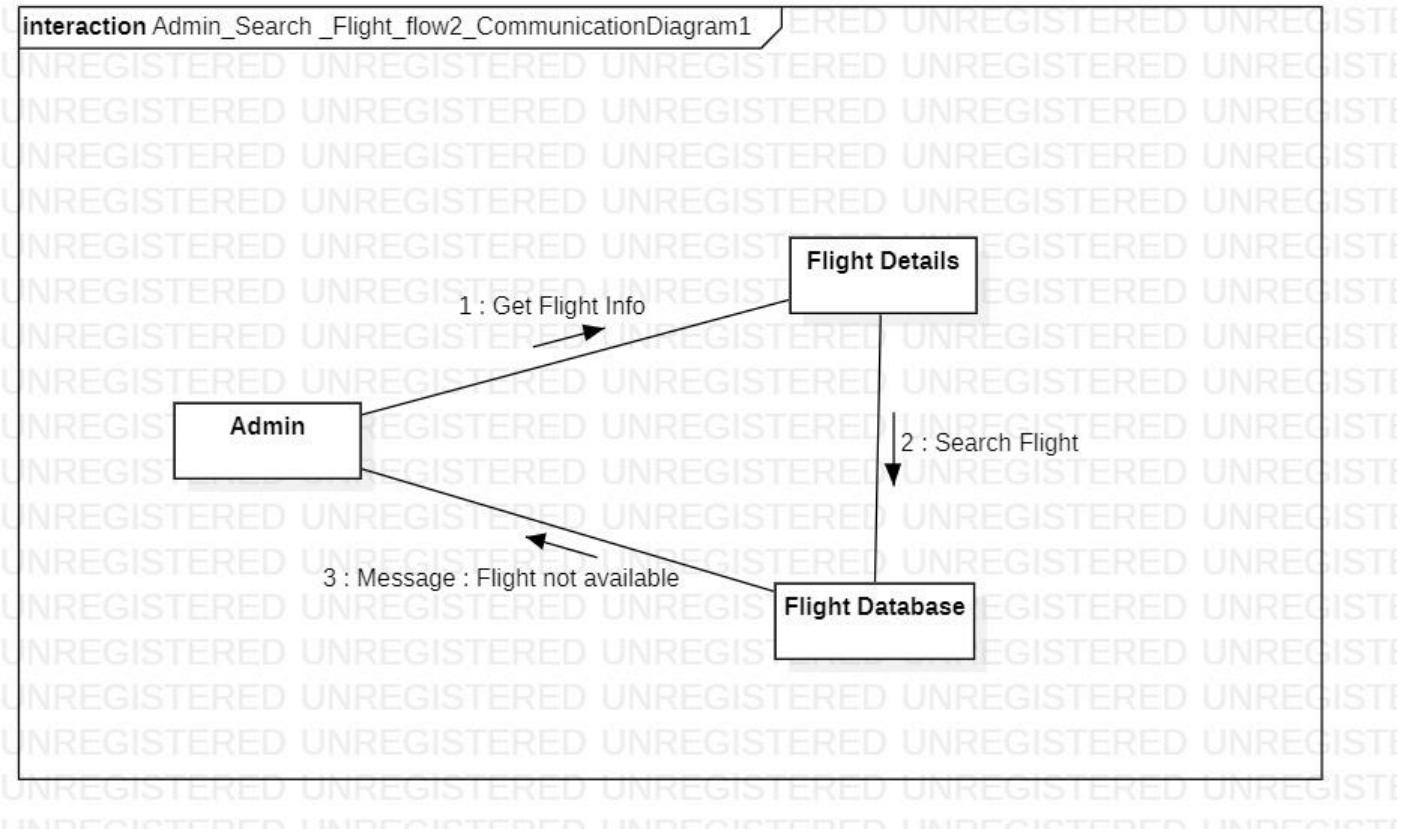
3. Add flight:



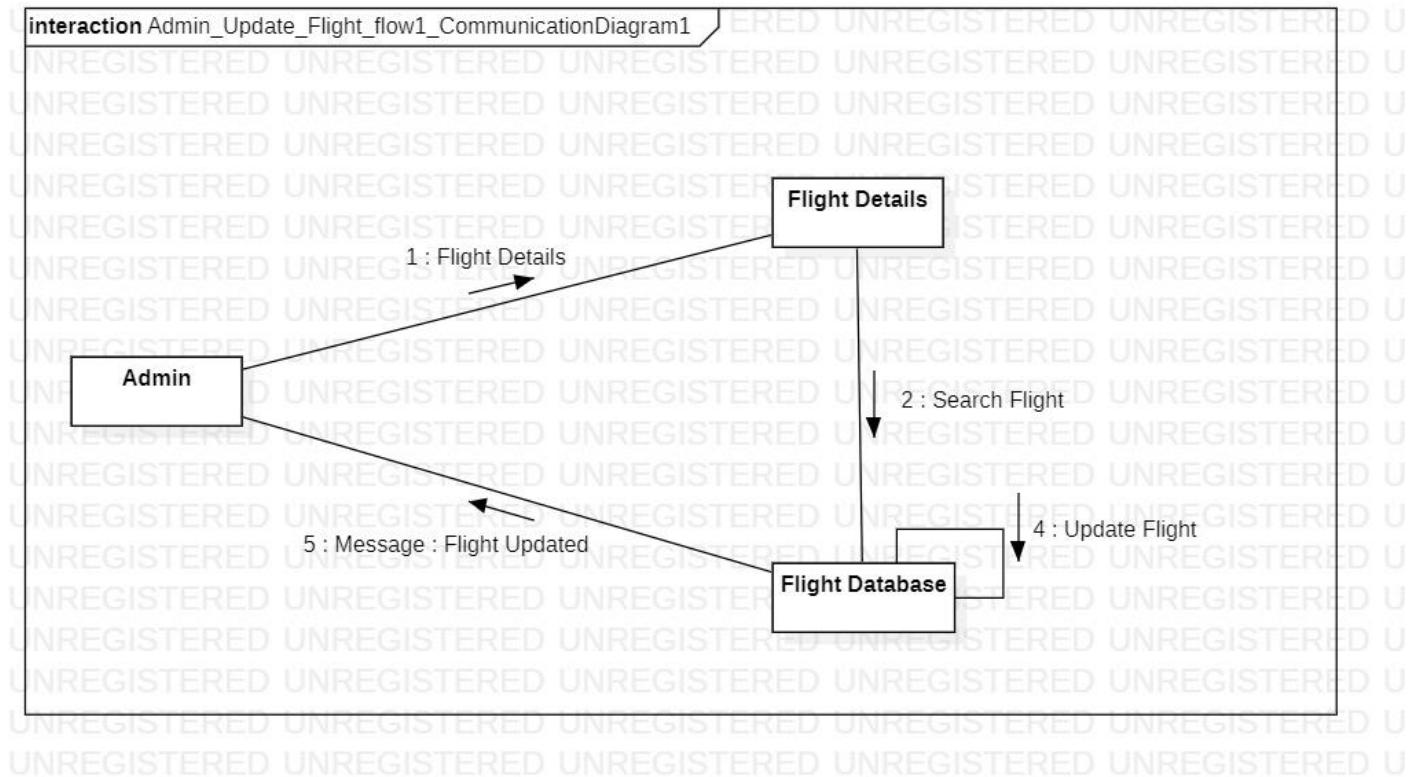
4. Search Flight flow 1:



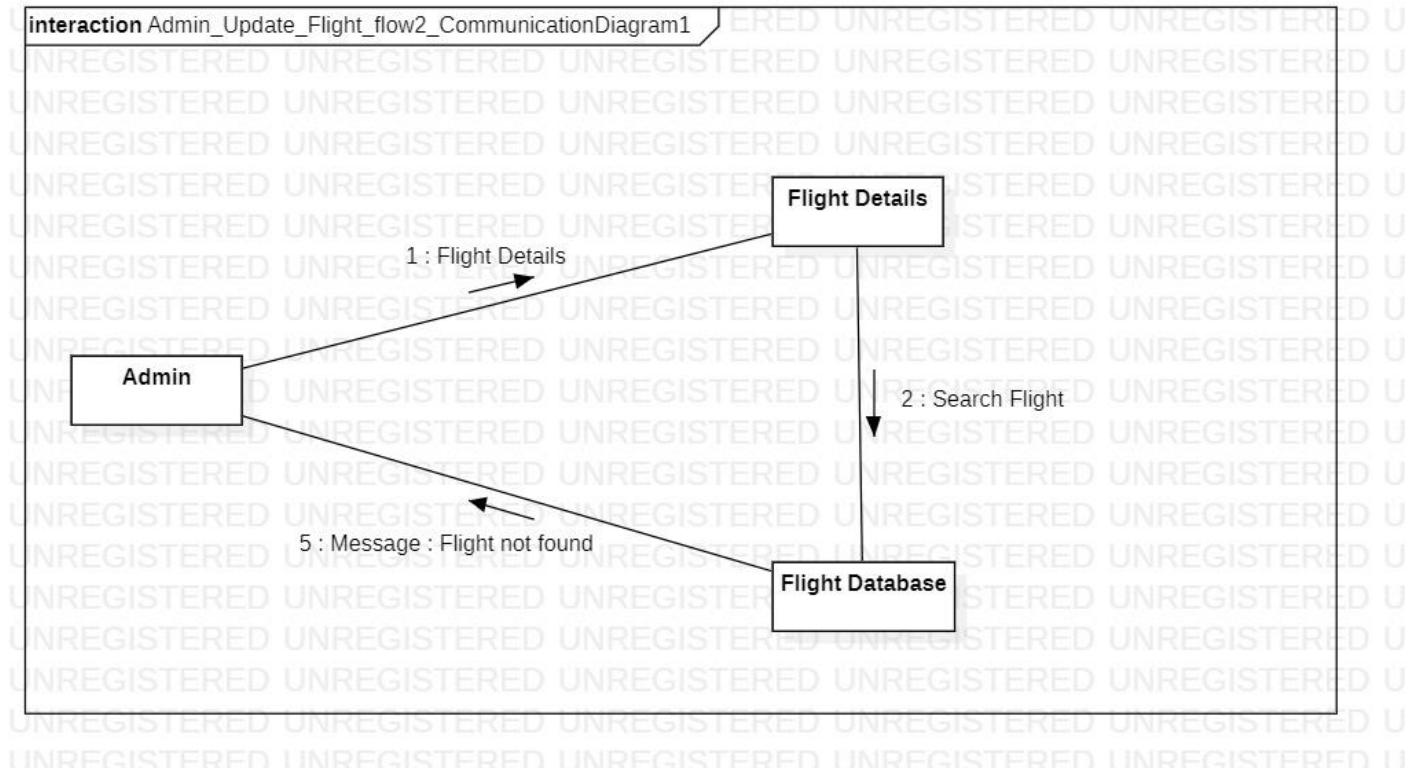
5. Search Flight flow 2:



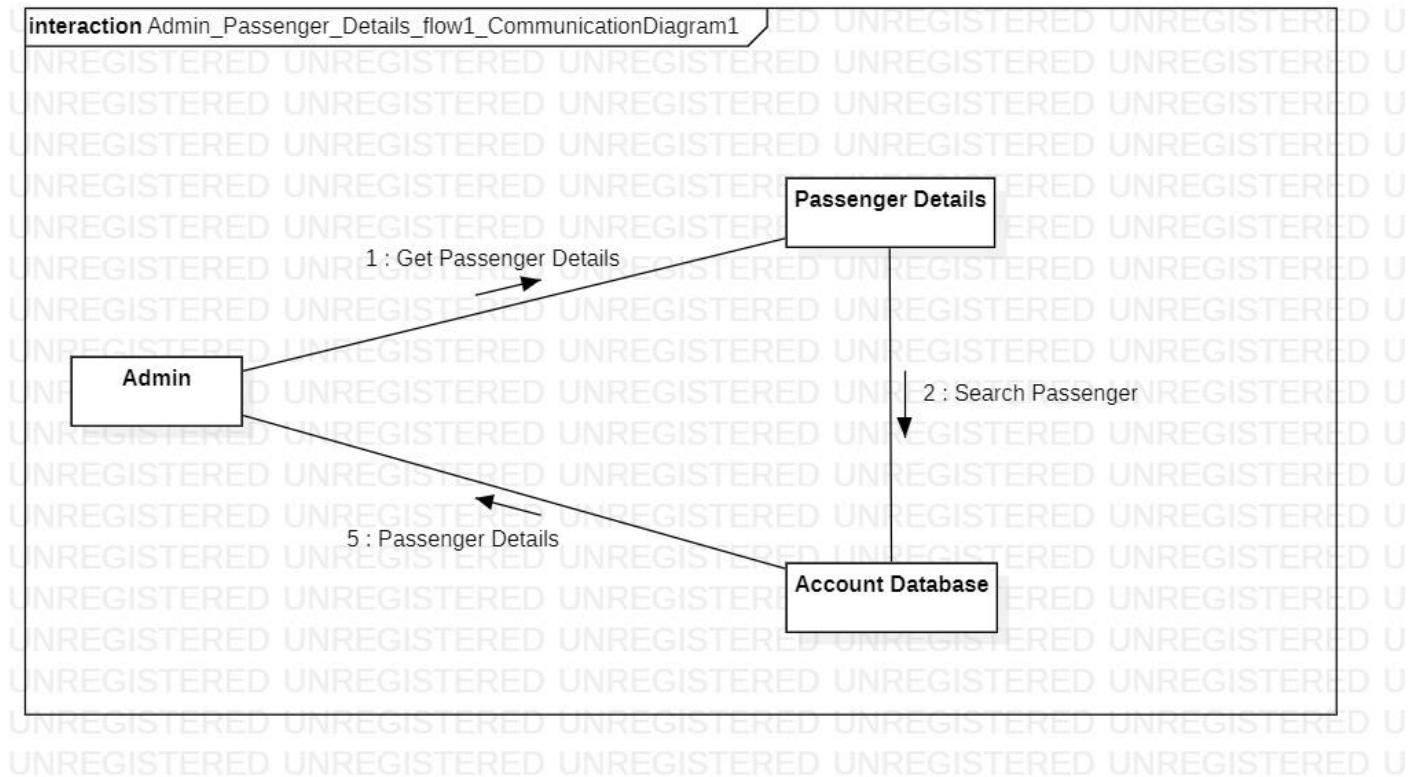
6. Update Flight flow1:



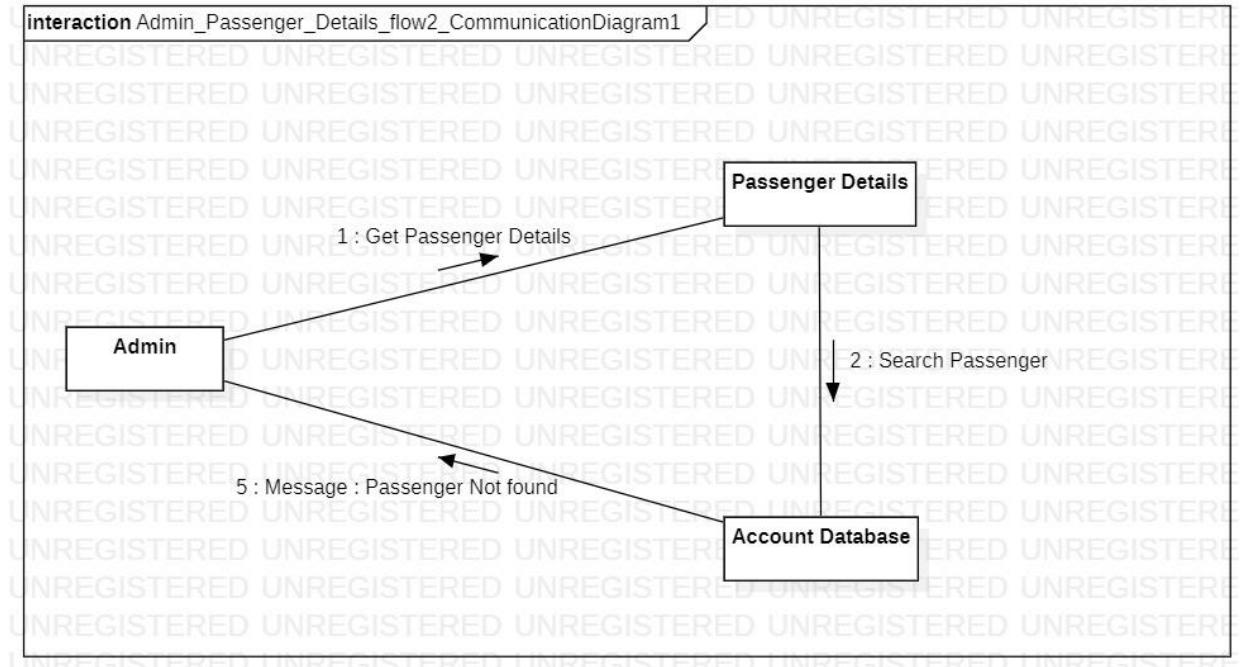
7. Update Flight flow 2:



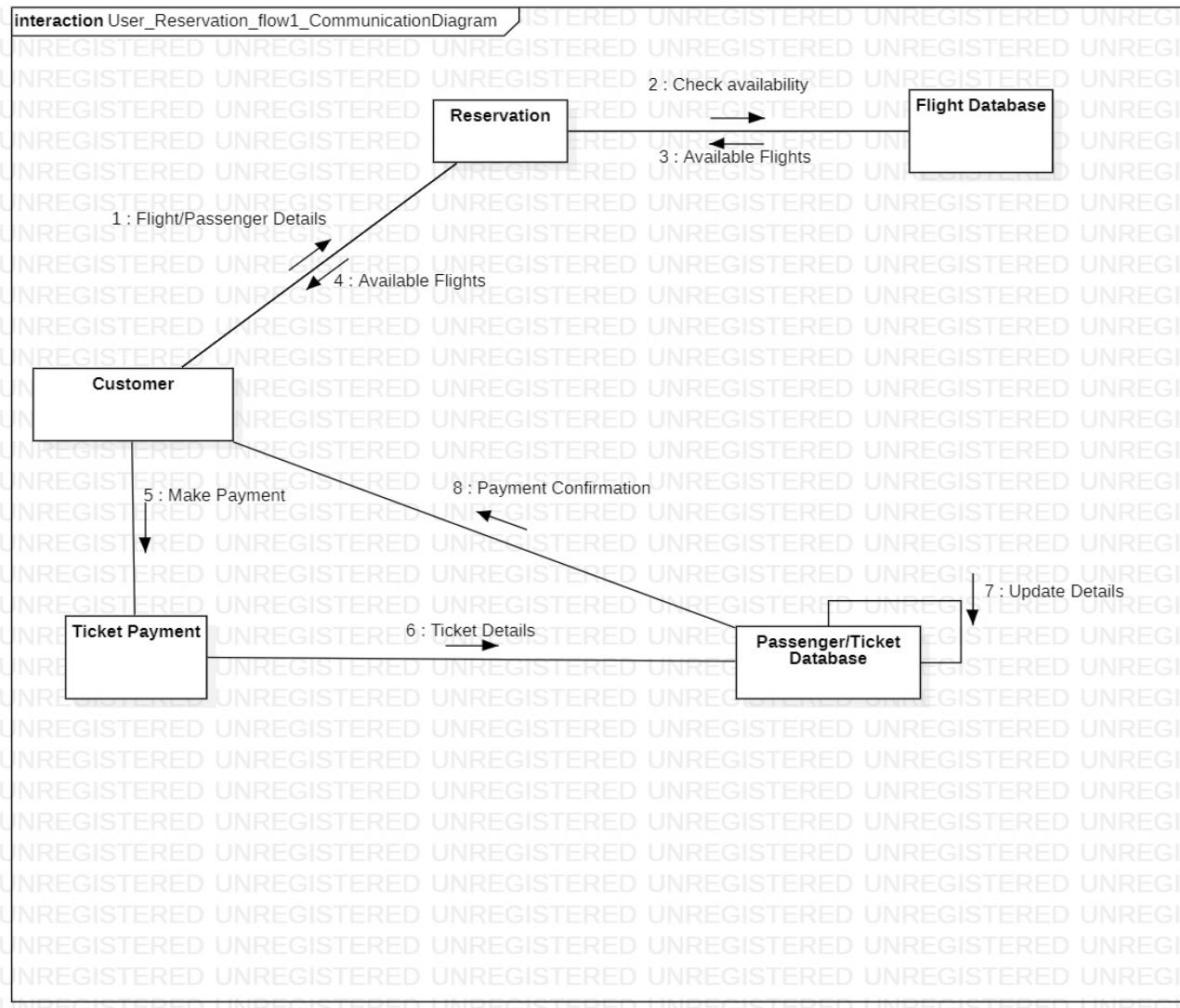
8. Get Passenger's details:



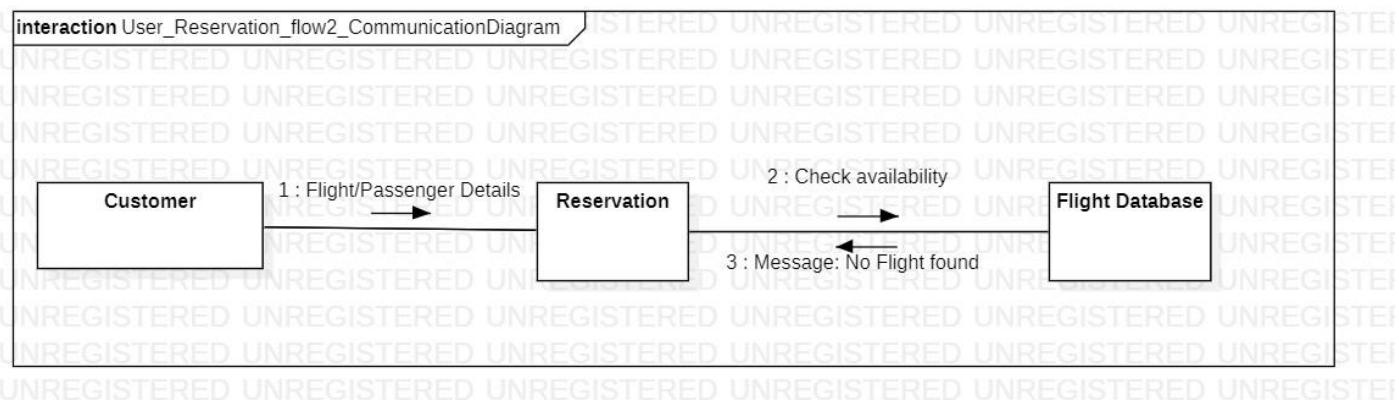
9. Get Passenger's details:



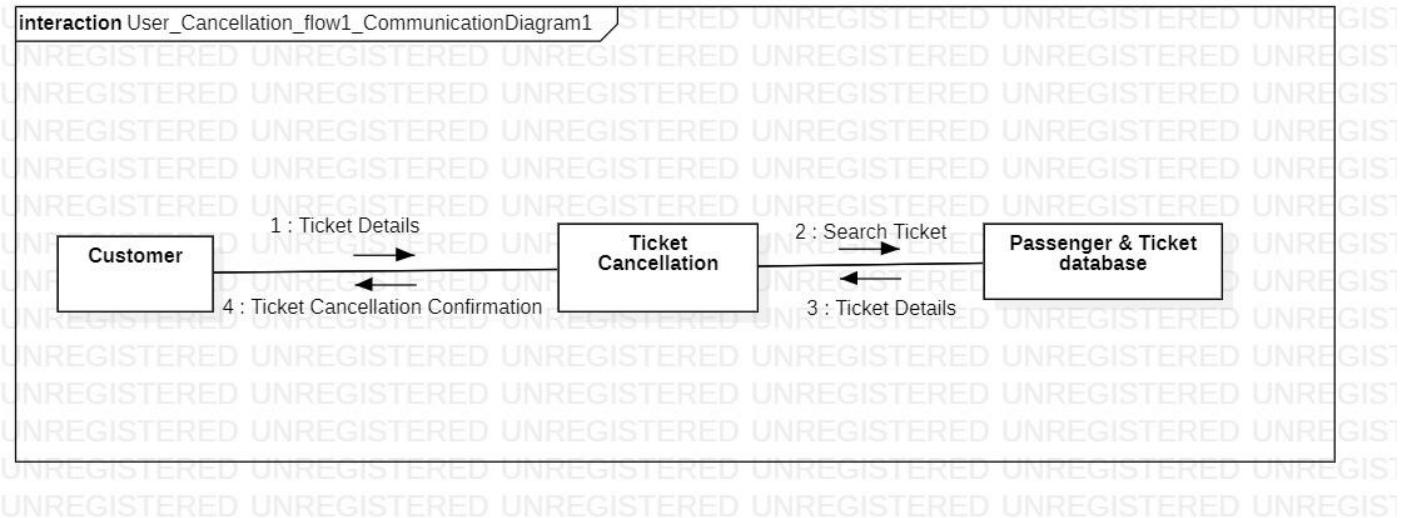
10. Book Ticket flow 1:



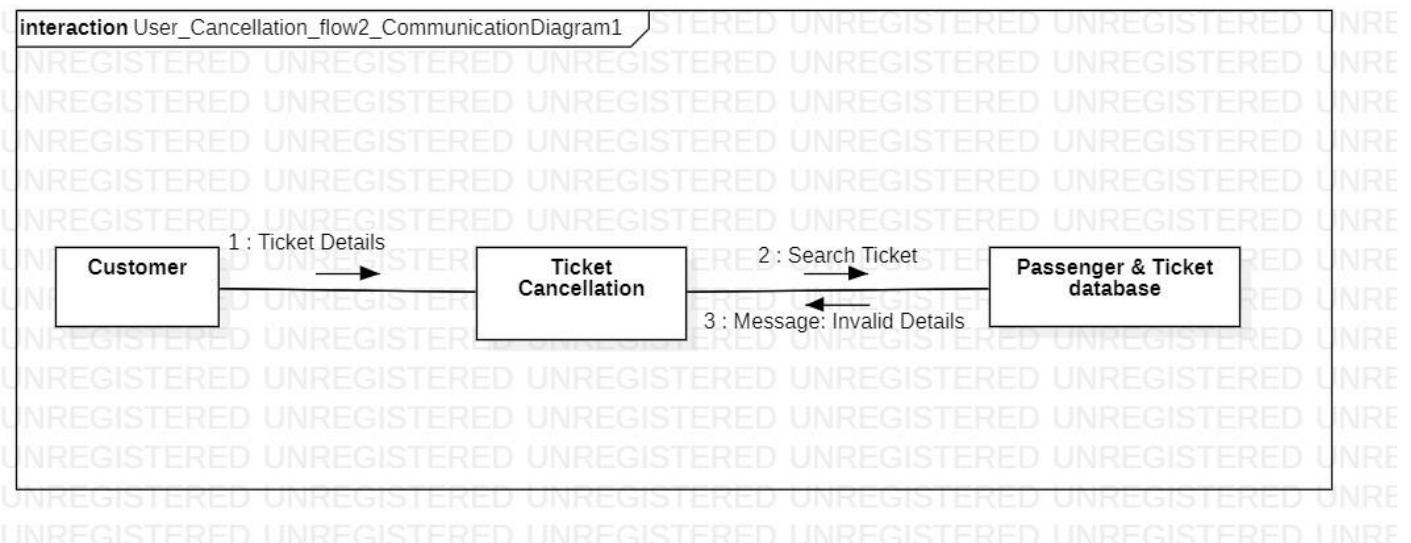
11. Book Ticket flow 2:



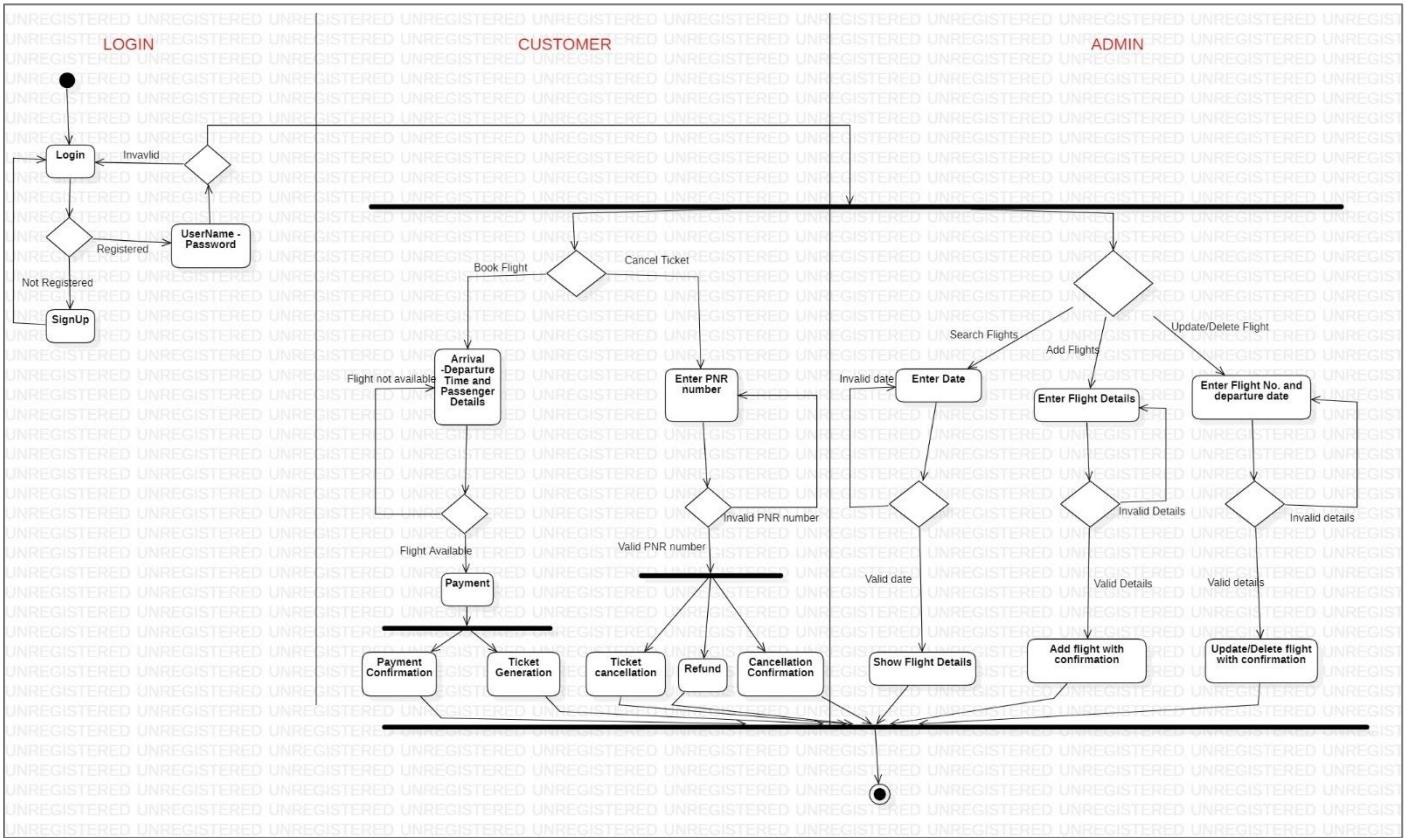
12. Ticket Cancellation flow 1:



13. Ticket Cancellation flow 2



ACTIVITY DIAGRAMS



TEST PLAN AND REPORT

OBJECTIVES

The main objectives of the test plan for the Airline Reservation System are as follows:

1. To identify the features of the system that will be tested.
2. To define the pass/fail criteria for each item that will be tested
3. To discuss the testing techniques being used to test the Airline Reservation System.

REFERENCES

The following references have been used in the preparation of the Test Plan document for the Airline Reservation System:

DEFINITIONS

The following are some of the terms and definitions that are related to the test plan of the Airline Reservation System:

- **Pass/Fail criteria:** Decision rules that are used to determine whether a software item passes or fails a test.
- **Test:** A collection of one or more test cases
- **Test Item:** A software item that is an objective of testing.
- **Test Plan:** A document describing the scope, approach, resources and schedule of the intended testing activities.
- **Test Summary Report:** A document summarizing the testing activities and results.
- **Testing:** The process of analysing a software item to detect the differences between the existing and required conditions.

ENVIRONMENTAL REQUIREMENTS

Hardware

- Working internet connection

Software

- The requirements for a client are:
 - Operating System - Windows/Linux
 - HTML/CSS
- The server should have:
 - Apache
 - PHP
 - MySQL

TEST ITEMS

This section of the test plan lists all the items of the Airline Reservation System project that will be tested:

- User Login
- New User Registration
- Search and book flights
- Tickets Cancelling
- View Flight Details
- View Passenger Details
- Add Flight
- Delete Flight

TEST CASES

The following are the test cases for the Airline Reservation System:

TEST CASE 1 – USER LOGIN

Incorrect Input	Incorrect user credentials (username/password/user type).
Pass Criteria	Message is generated to indicate that an invalid user credentials have been typed.
Correct Input	Valid user credentials are entered.
Pass Criteria	User should be directed to the webpage that he is intended to go to after he logs into the system.

Photo 1: Invalid Credentials

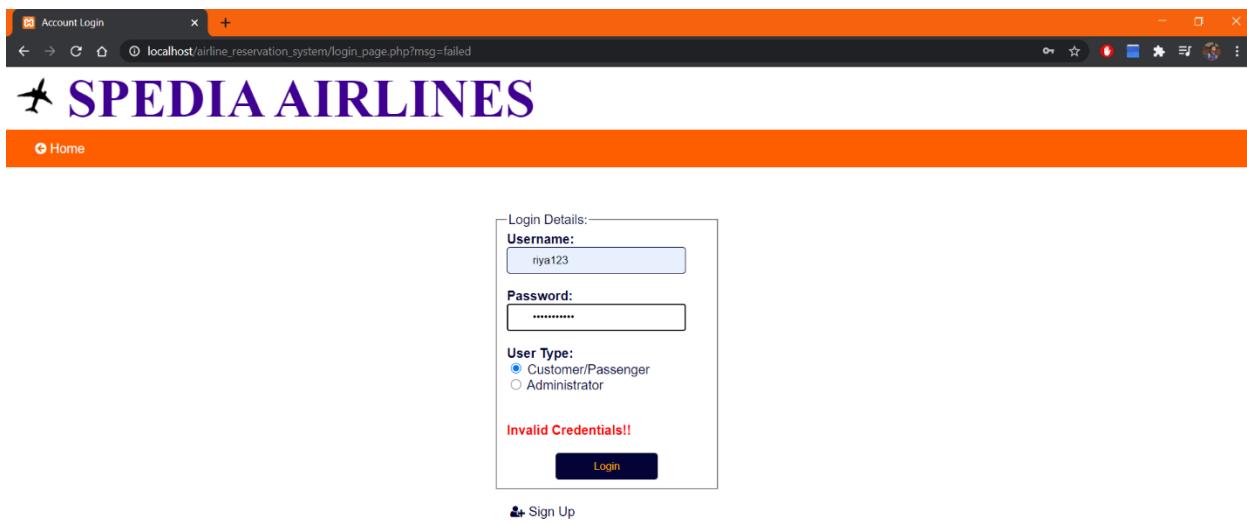
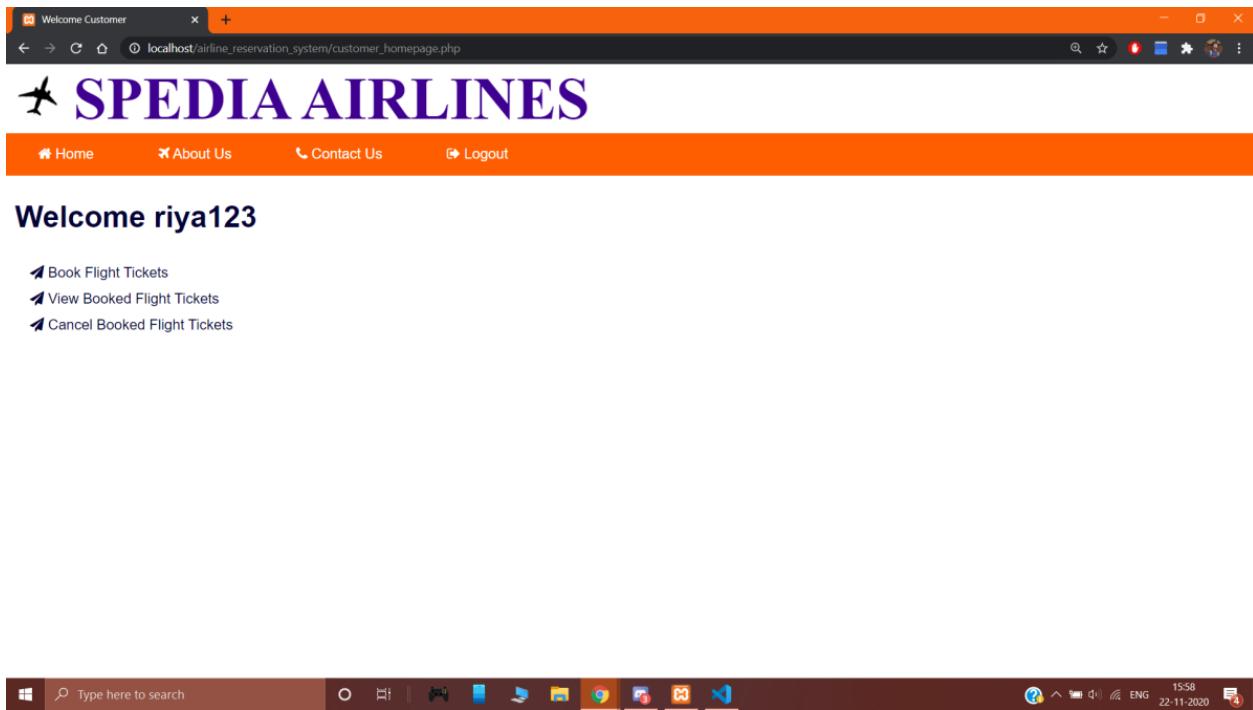


Photo 2: Correct Credentials & User is able to login



TEST CASE 2 – USER LOGIN

Incorrect Input	Login fields are left empty in the login page.
Pass Criteria	Appropriate message is generated to indicate that the login fields are left empty.
Correct Input	Every field is filled in the login page.
Pass Criteria	Either the user is logged in or an error message is shown depending on the validity of the user credentials.

Photo 3: Empty field error

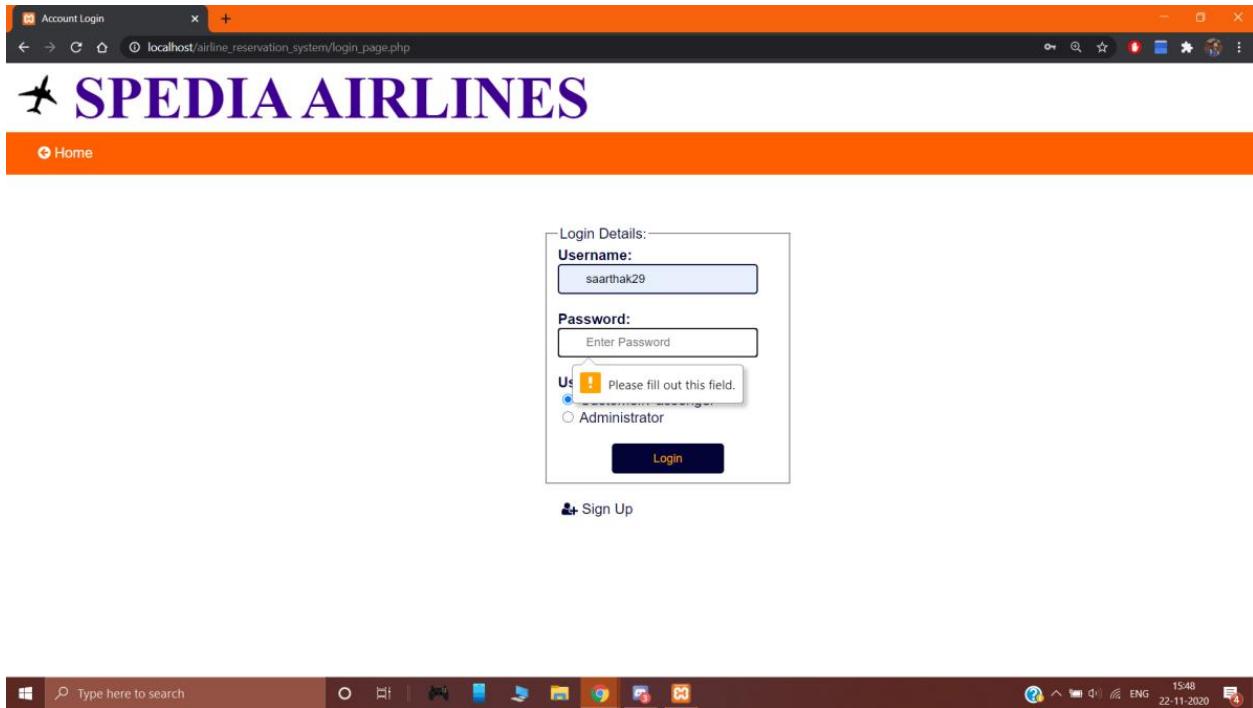
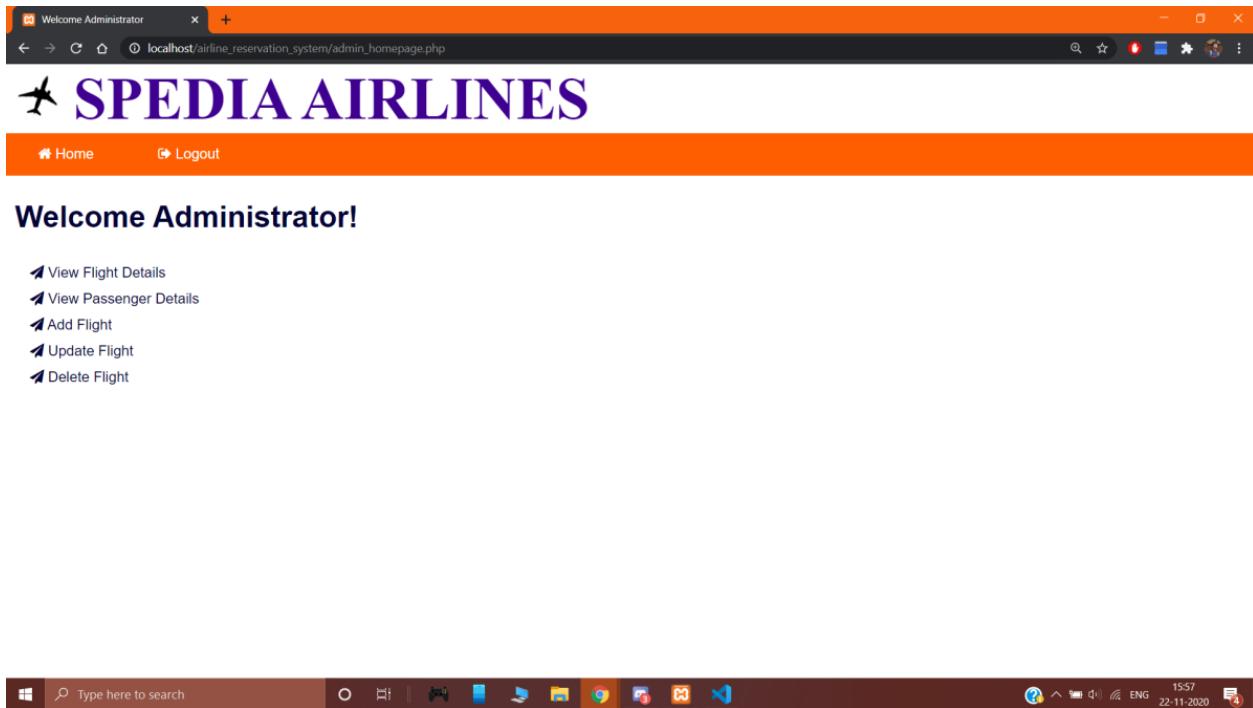


Photo 4: User logged in when every field is filled (Correctly)



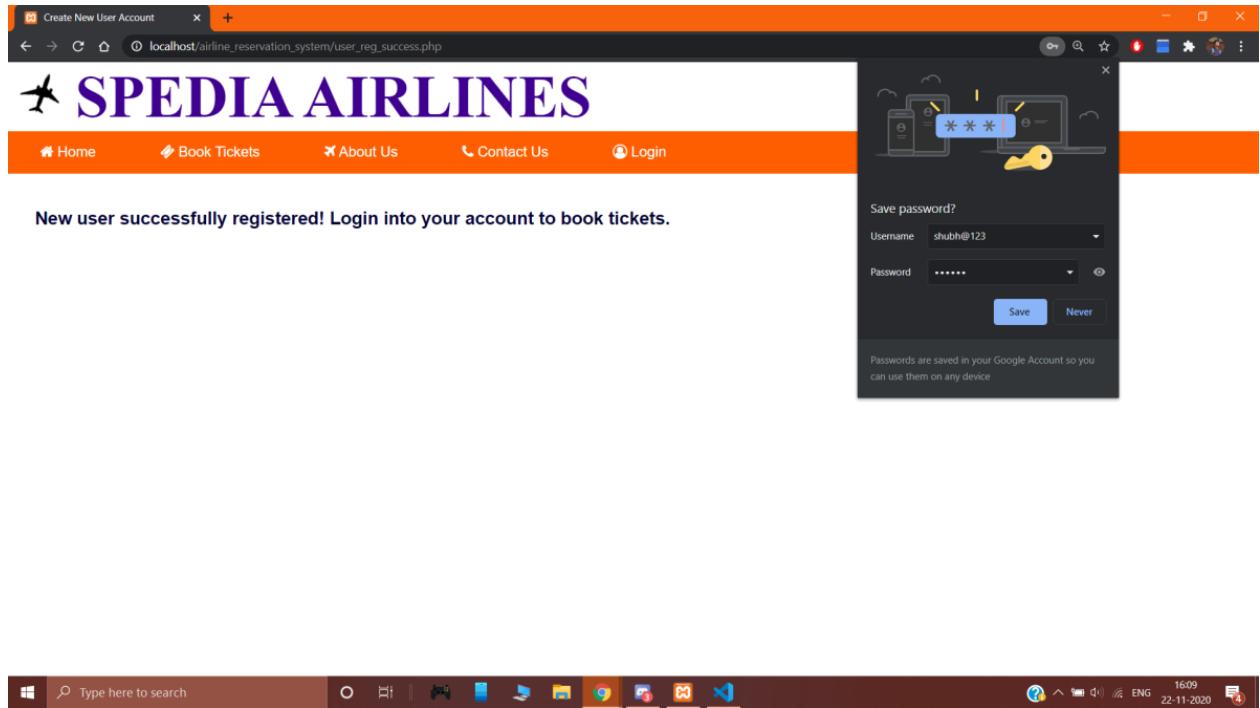
TEST CASE 3 – NEW USER REGISTRATION

Incorrect Input	The data fields are left out empty in the registration page.
Pass Criteria	An error message is generated to the user indicating that the fields are to be filled in order to be registered into the system.
Correct Input	The customer enters the data in all the fields in the registration form.
Pass Criteria	It accepts all the customer details and then registers the customer and helps him log into the system.

Photo 5: Empty field error

The screenshot shows a web browser window titled "Create New User Account" with the URL "localhost/airline_reservation_system/new_user.php". The page has two main sections: "ENTER LOGIN DETAILS" and "ENTER CUSTOMER'S PERSONAL DETAILS". In the "ENTER CUSTOMER'S PERSONAL DETAILS" section, the "Enter your email ID" field is empty and highlighted with a red box. An error message "Please fill out this field." is displayed below it. The rest of the fields in this section contain valid data: "Enter your name" (Shubham Sharma), "Enter your phone no." (9810067338), and "Enter your address" (K-4/55 Rohini West-2). A "Submit" button is at the bottom.

Photo 6: User signed-up when every field is filled



Test Case 4 - Search and Book Flights

Incorrect Input	The data fields are left out empty in the book flights page.
Pass Criteria	An error message is generated to the user indicating that the fields are to be filled in order to book desired tickets.
Correct Input	The customer enters the data in all the fields.
Pass Criteria	Either it would show valid results or an error message depending on the validity of the data.

Photo 7: Empty field error

The screenshot shows a web browser window for 'View Available Flights' at 'localhost/airline_reservation_system/book_tickets.php'. The title bar says 'View Available Flights'. The main content area has a purple header 'SPEDIA AIRLINES' with a plane icon. Below it is an orange navigation bar with 'Home', 'Dashboard', and 'Logout' links. The main form is titled 'SEARCH FOR AVAILABLE FLIGHTS'. It has four input fields: 'Enter the Origin' (Gurgaon), 'Enter the Destination' (Hyderabad), 'Enter the Departure Date' (dd-mm-yyyy), and 'Enter the No. of Passengers' (Eg. 5). A tooltip 'Please fill out this field.' appears over the date field. A dropdown menu for class ('Economy') is open. At the bottom is a dark blue button labeled 'Search for Available Flights'. The taskbar at the bottom shows the Windows logo, a search bar with 'Type here to search', and various pinned icons like File Explorer, Edge, and File History. The system tray shows the date '22-11-2020' and time '16:10'.

Test Case 5 - Search and Book Flights

Incorrect Input	Incorrect format of data entered into the data entry fields.
Pass Criteria	A message is generated indicating the wrong format of data is entered.
Correct Input	Correct format of input is entered.
Pass Criteria	The search would return results based on the search parameters after which the booking process is continued.

Photo 8: Incorrect data format (Date)

The screenshot shows a web browser window for 'View Available Flights' at 'localhost/airline_reservation_system/book_tickets.php'. The title bar says 'View Available Flights'. The main header features the 'SPEDIA AIRLINES' logo. A navigation bar includes links for 'Home', 'Dashboard', and 'Logout'. Below the header, the section title 'SEARCH FOR AVAILABLE FLIGHTS' is displayed. The form fields include 'Enter the Origin' (Gurgaon), 'Enter the Destination' (Hyderabad), 'Enter the Departure Date' (01-02-2020), and 'Enter the No. of Passengers' (5). A validation error message 'Value must be 22-11-2020 or later.' is shown next to the date field. A dropdown menu for 'Business' class is open. At the bottom is a dark blue button labeled 'Search for Available Flights'. The taskbar at the bottom of the screen shows the Windows Start button, a search bar with 'Type here to search', and various pinned icons like File Explorer, Edge, and Mail. The system tray shows the date '22-11-2020' and time '16:11'.

Test Case 6 - Search and Book Flights

Incorrect Input	Incorrect search parameters entered.
Pass Criteria	A message is generated indicating no information is available for the incorrect search parameters.
Correct Input	Correct search parameters are entered.
Pass Criteria	The search would return valid results and then after the booking process, the system would generate a PNR number for the customer which is saved in the account.

Photo 9: Incorrect flight search parameters

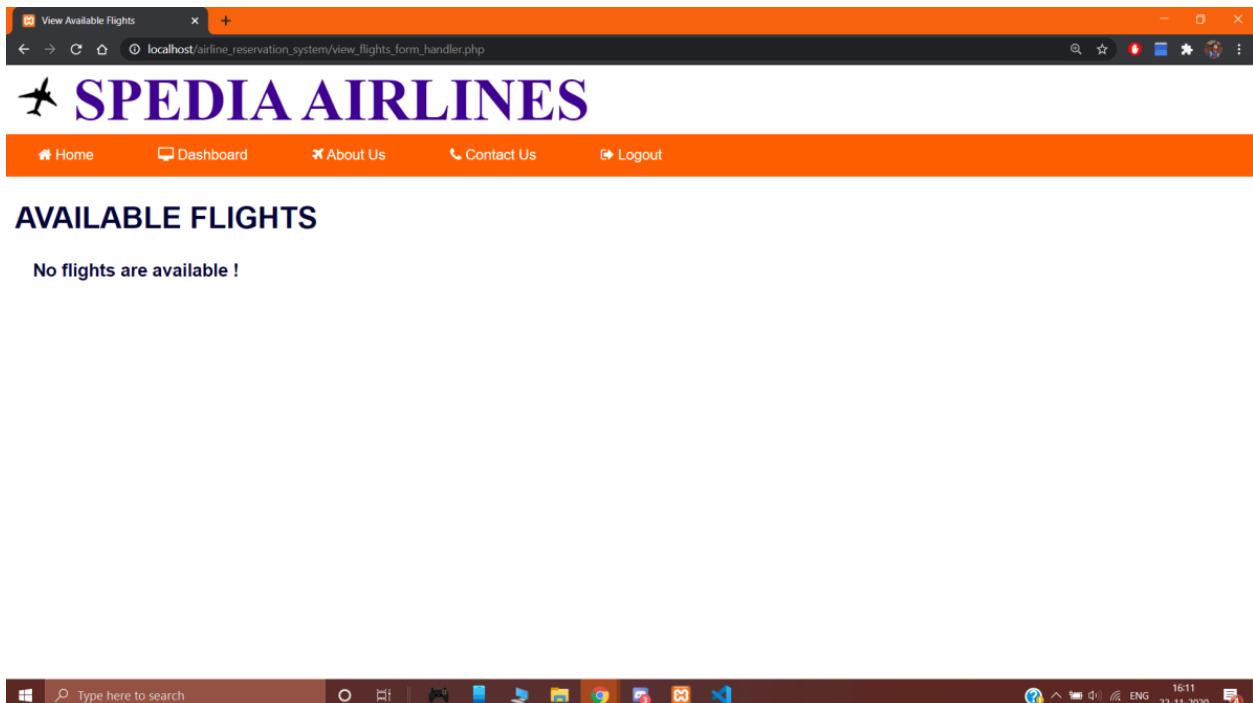


Photo 10: Desired flight results displayed after correct flight search parameters are entered

The screenshot shows a web browser window with the title "View Available Flights" and the URL "localhost/airline_reservation_system/view_flights_form_handler.php". The page header features the "SPEDIA AIRLINES" logo with a stylized airplane icon. Below the header is a navigation bar with links for Home, Dashboard, About Us, Contact Us, and Logout. The main content area is titled "AVAILABLE FLIGHTS" and displays a table of flight results:

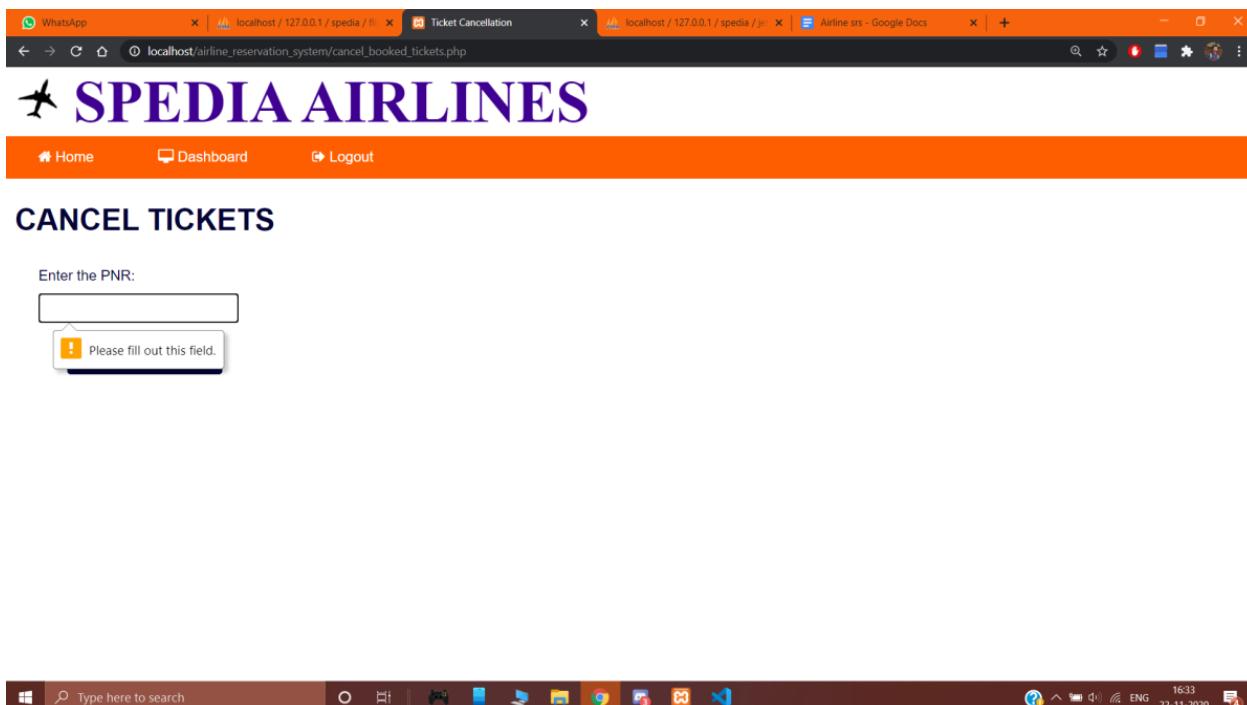
Flight No.	Origin	Destination	Departure Date	Departure Time	Arrival Date	Arrival Time	Price(Business)	Select
FL156	New Delhi	Chennai	2020-12-01	05:00:00	2020-12-01	09:00:00	₹ 5750	<input type="radio"/>
FI690	New Delhi	Chennai	2020-12-01	12:00:00	2020-12-01	17:00:00	₹ 5950	<input type="radio"/>

A blue "Select Flight" button is located at the bottom of the table. The browser's taskbar at the bottom shows various pinned icons and the system clock indicating 16:19 on 22-11-2020.

Test Case 7 - Tickets Cancelling

Incorrect Input	The PNR field is left out empty.
Pass Criteria	An error message is generated to the user saying that they have to fill out that field in order to cancel flight.
Correct Input	The customer enters the PNR.
Pass Criteria	Either it would show valid results or an error message depending on the validity of the PNR.

Photo 11: Empty field error



Test Case 8 - Tickets Cancelling

Incorrect Input	Customer enters invalid PNR number.
Pass Criteria	An error message is shown informing the customer about the invalid PNR number.
Correct Input	Customer enters valid PNR number.
Pass Criteria	A message is shown that the tickets is cancelled successfully and the ticket amount would be refunded to his account with 15% cancellation charges.

Photo 12: Invalid PNR number

The screenshot shows a Microsoft Edge browser window with the following details:

- Address bar: localhost/airline_reservation_system/cancel_booked_tickets.php?msg=failed
- Title bar: SPEDIA AIRLINES
- Header: Home, Dashboard, Logout
- Main Content:
 - CANCEL TICKETS**
 - *Invalid PNR, please enter PNR again
 - Enter the PNR:
 - Cancel Ticket
- Taskbar at the bottom: Type here to search, Start button, Task View, File Explorer, Control Panel, Google Chrome, FileZilla, and File Explorer icons. System tray shows battery level, signal strength, ENG, 16:33, 22-11-2020, and a notification icon.

Photo 13: Passenger's ticket is cancelled after correct PNR number is entered

The screenshot shows a web browser window with four tabs open:

- WhatsApp
- localhost / 127.0.0.1 / spedia /
- Cancel Booked Tickets
- localhost / 127.0.0.1 / spedia /

The main content area displays the SPEDIA AIRLINES logo and navigation links for Home, Dashboard, and Logout. Below this, a section titled "CANCEL BOOKED TICKETS" contains the following messages:

Your ticket has been cancelled successfully.
Your amount of ₹ 6851.00 will be refunded to your bank account (Cancellation charge on 15% of your ticket amount has been deducted).

The browser's address bar shows the URL: localhost/airline_reservation_system/cancel_booked_tickets_success.php.

Test Case 9 - View Flight Details

Input	A date on which there is/was no flight available.
Pass Criteria	"No flights to show" message is shown to the user.
Input	A valid date on which there is/was flight(s) available.
Pass Criteria	Details of flight(s) available on that date are shown to the user.

Photo 14: View flight details page

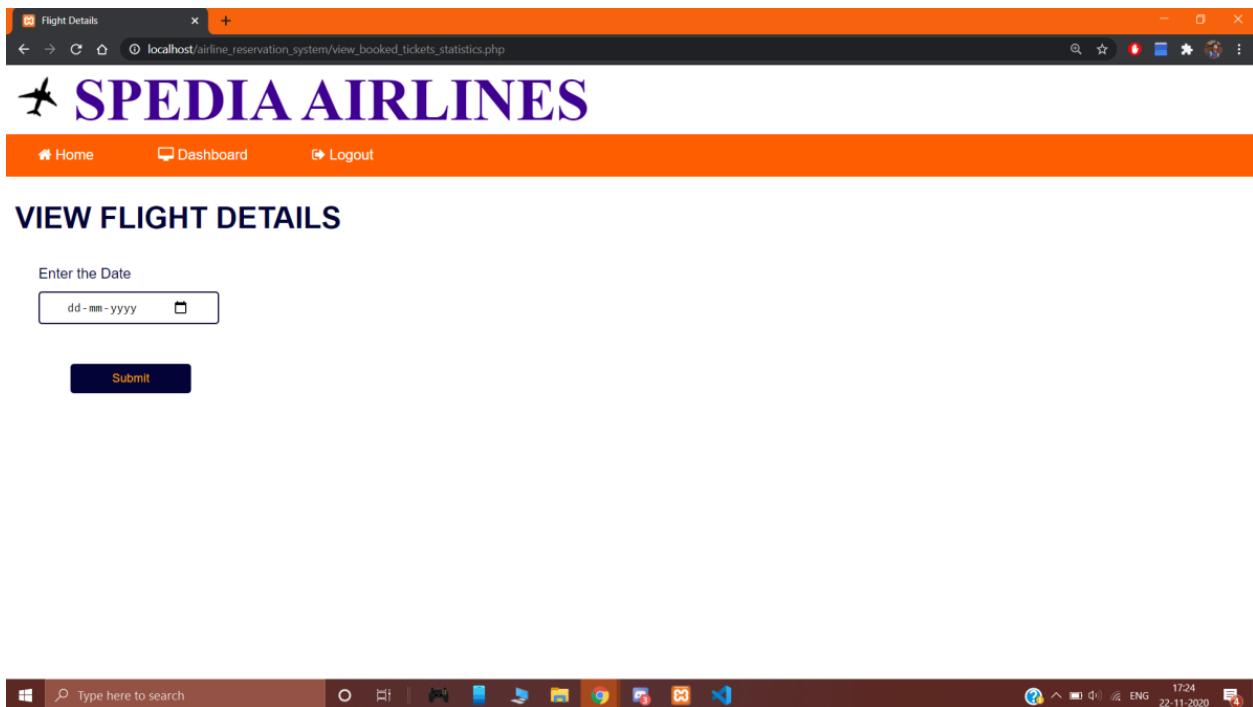


Photo 15: Incorrect flight search parameters

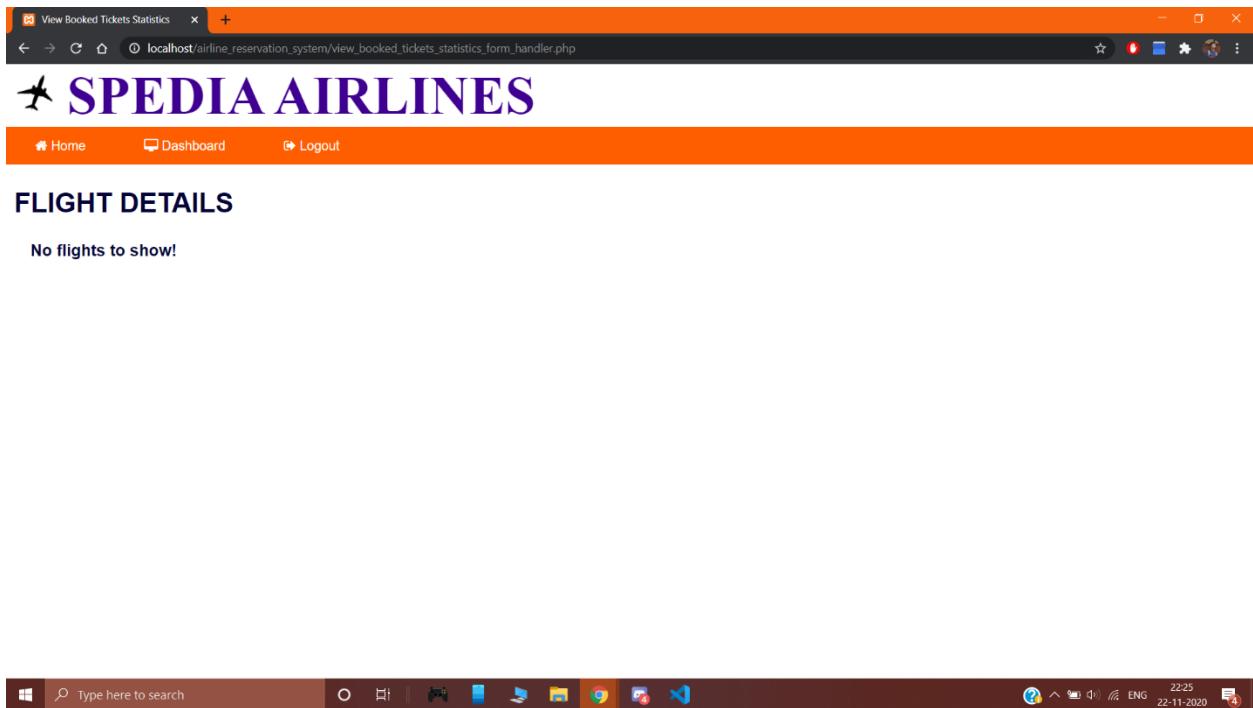
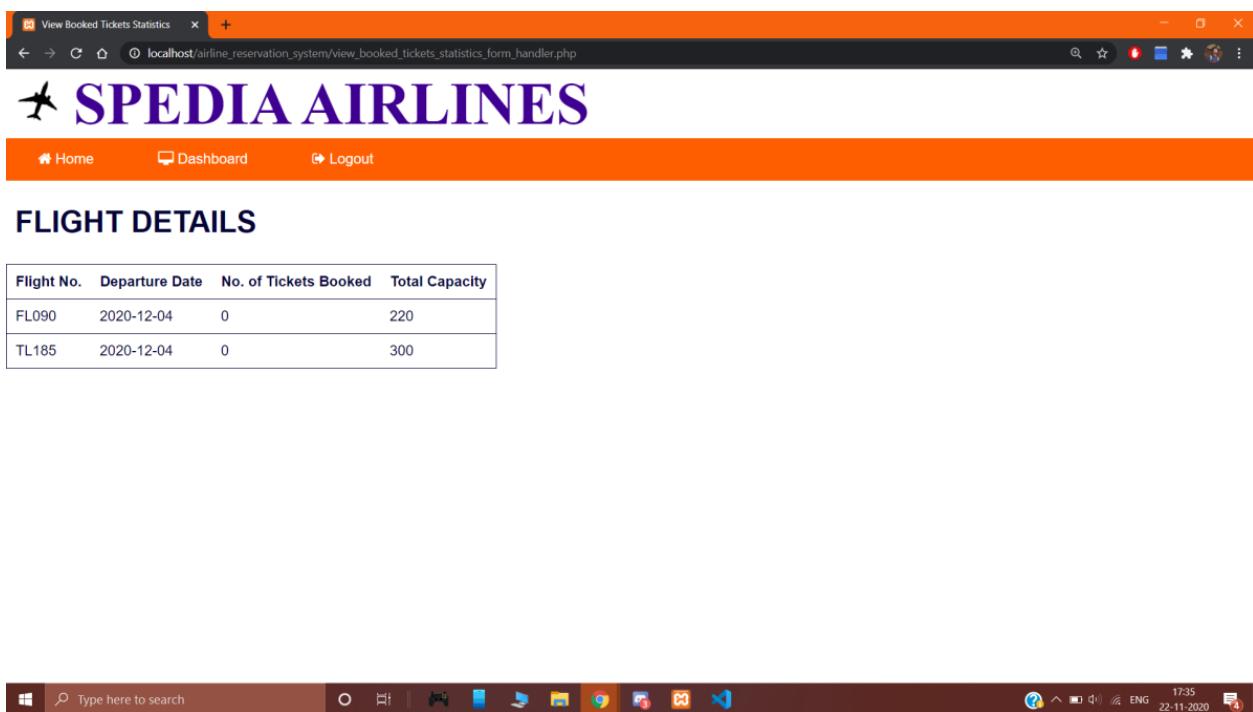


Photo 16: Desired flight details displayed after correct search details



Test Case 10 - Details of Passengers

Incorrect Input	User inputs invalid Flight No. or Date.
Pass Criteria	'No information is available' message is shown to the user.
Correct Input	User inputs valid Flight No. and date.
Pass Criteria	Details of passengers for that particular flight is shown to the user.

Photo 17: View passenger details page

The screenshot shows a web browser window titled "Passenger Details". The URL in the address bar is "localhost/airline_reservation_system/admin_view_booked_tickets.php". The page has a header with the "SPEDIA AIRLINES" logo and navigation links for "Home", "Dashboard", and "Logout". The main content area is titled "VIEW THE PASSENGER DETAILS FOR A FLIGHT". It contains two input fields: "Enter the Flight No." and "Enter the Departure Date". Below these fields is a "Submit" button. The browser's taskbar at the bottom shows various pinned icons and the system clock indicating 17:41 on 22-11-2020.

Photo 18: Incorrect search details

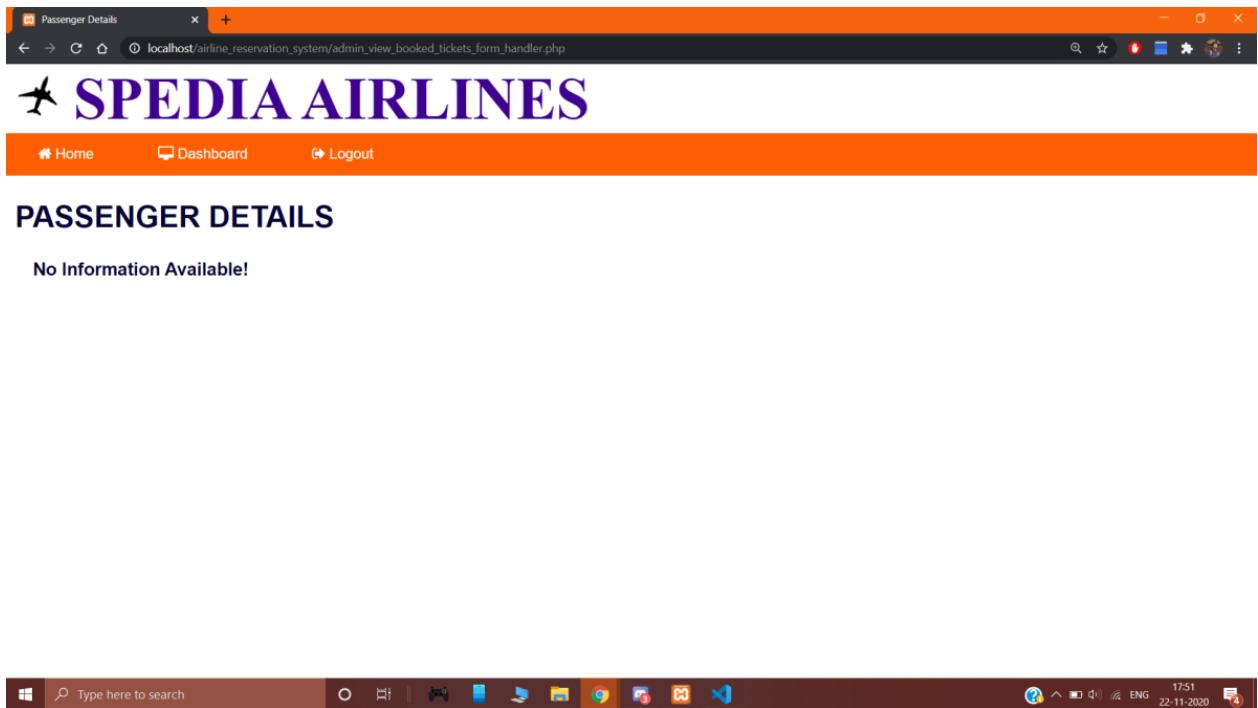
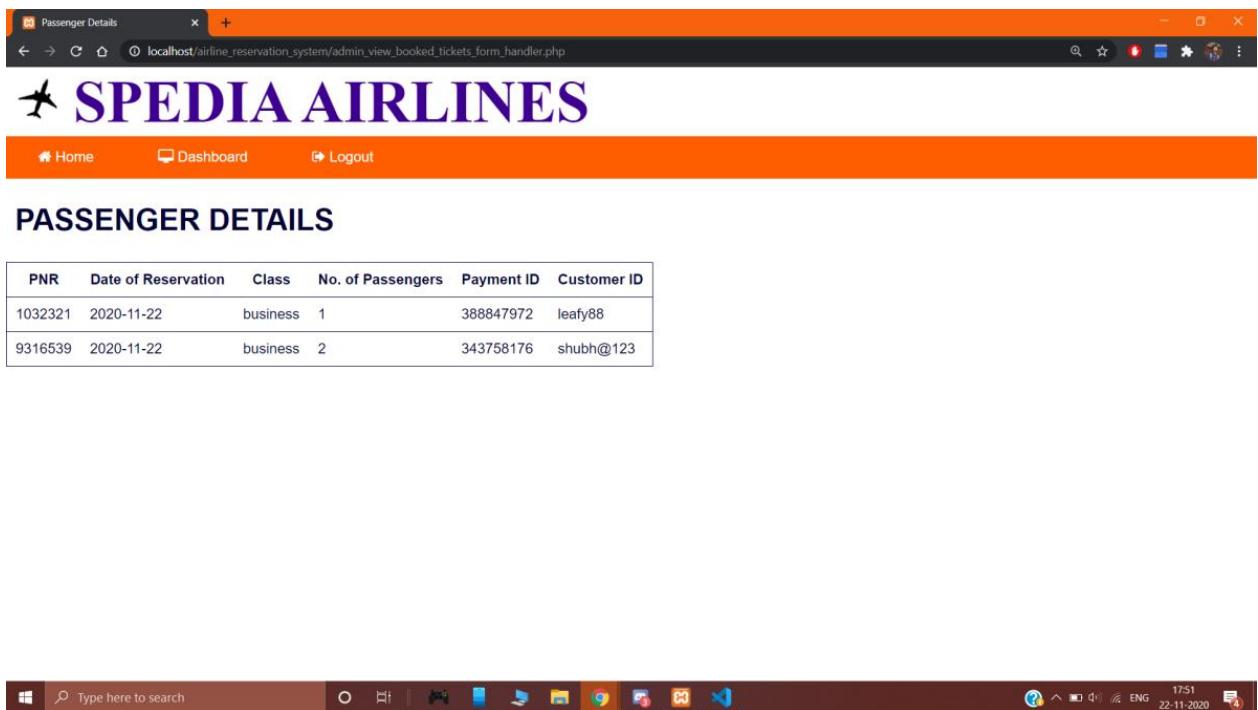


Photo 19: Desired passenger details displayed after correct search details



Test Case 11 - Add Flights

Incorrect Input	The data fields are left out empty in the Add Flights page.
Pass Criteria	An error message is generated to the user indicating that the fields are to be filled in order to add desired flight.
Correct Input	The user enters the appropriate data in all the fields.
Pass Criteria	A message is shown to the user informing about successful flight addition.

Photo 20: Empty field error

ENTER THE FLIGHT SCHEDULE DETAILS

Flight Number
TR098

Origin Gurgaon Destination Dubai

Departure Date 03-12-2020 Arrival Date 04-12-2020

Departure Time 20:00 Arrival Time 00:00

Number of Seats in Economy Class 100 Number of Seats in Business Class 50

Ticket Price(Economy Class) 8000 Ticket Price(Business Class) 10000

Jet ID

Please fill out this field.

Photo 21: Flight successfully added after appropriate details are filled in every field

The screenshot shows a web browser window titled "Add Flight Schedule Details" with the URL "localhost/airline_reservation_system/add_flight_details.php?msg=success". The page header features the "SPEDIA AIRLINES" logo. Below the header, there are navigation links for "Home", "Dashboard", and "Logout". The main content area is titled "ENTER THE FLIGHT SCHEDULE DETAILS". A green success message states "The Flight Schedule has been successfully added.". The form fields include: "Flight Number" (empty input box), "Origin" (empty input box) and "Destination" (empty input box), "Departure Date" (date input box with placeholder "dd-mm-yyyy") and "Arrival Date" (date input box with placeholder "dd-mm-yyyy"), "Departure Time" (time input box with placeholder "-- : --") and "Arrival Time" (time input box with placeholder "-- : --"). At the bottom of the screen, the Windows taskbar is visible with various icons and the system tray showing the date and time as "22-11-2020 17:54".

Test Case 12 - Delete Flights

Incorrect Input	User inputs invalid Flight No. or Date of Departure.
Pass Criteria	An error message is shown to the user indicating that the flight no./date is invalid.
Correct Input	User inputs valid flight no. and date of departure.
Pass Criteria	A message is shown to the user informing about successful flight deletion.

Photo 21: Invalid flight details

The screenshot shows a web browser window titled "Delete Flight Schedule Details". The URL in the address bar is "localhost/airline_reservation_system/delete_flight_details.php?msg=failed". The page header features the "SPEDIA AIRLINES" logo. Below the header, there is a navigation bar with links for "Home", "Dashboard", and "Logout". The main content area has a heading "ENTER THE FLIGHT SCHEDULE TO BE DELETED". A red error message "Invalid Flight No./Departure Date, please enter again." is displayed above two input fields. The first input field is labeled "Enter a valid Flight No." and the second is labeled "Enter the Departure Date". Both fields are empty. At the bottom of the form is a blue "Delete" button. The browser's taskbar at the bottom shows various pinned icons and the system tray indicates the date as 22-11-2020 and time as 17:59.

Photo 22: Flight is deleted after correct details are entered

The screenshot shows a web browser window titled "Delete Flight Schedule Details" with the URL "localhost/airline_reservation_system/delete_flight_details.php?msg=success". The page header features the "SPEDIA AIRLINES" logo. Below the header, there are navigation links for "Home", "Dashboard", and "Logout". A main heading "ENTER THE FLIGHT SCHEDULE TO BE DELETED" is displayed. A success message "The Flight Schedule has been successfully deleted." is shown above two input fields: "Enter a valid Flight No." and "Enter the Departure Date". A "Delete" button is located below the date field. The Windows taskbar at the bottom of the screen includes icons for Start, Search, Task View, File Explorer, Control Panel, Mail, Photos, File History, Google Chrome, Facebook, and Microsoft Edge. System status icons show battery level, signal strength, and the date/time (22-11-2020, 18:02).

Scenario Matrix

Use case- User login, Registration and Deletion

Variables

1. Username/Email Id
2. Password
3. Name
4. DOB
5. Phone No.
6. Email ID
7. OTP
8. Deletion Confirmed

Input States

Valid or Invalid or N/A

Test Case Matrix for the above Use Case

Test Case ID	Scenario and Description	Input 1 : Username / Email ID	Input 2 : Password	Input 3 : User Name	Input 4 : DOB	Input 5 : Phone No.	Input 6 : Email ID	Input 7 : OTP	Deletion confirmed	Expected Result
TC 1	Scenario 1 - User Login	Valid	Valid	n/a	n/a	n/a	n/a	Valid	n/a	User is Logged In to the main page
TC 2	Scenario 2 - User Login Alternate Flow Invalid Entry	Invalid	Valid	n/a	n/a	n/a	n/a	Valid	n/a	Invalid Username / Email ID
TC 3		Valid	Invalid	n/a	n/a	n/a	n/a	Valid	n/a	Invalid Password
TC 4		Valid	Valid	n/a	n/a	n/a	n/a	Invalid	n/a	Invalid OTP
TC 5	Scenario 3 - User Login Alternate Flow User Not Found	Valid/Invalid	Valid/Invalid	n/a	n/a	n/a	n/a	n/a	n/a	User does not exist
TC 6	Scenario 4 - User Login Alternate Flow User Exits	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	User exits
TC 7	Scenario 5 - New User Addition	n/a	Valid	Valid	Valid	Valid	Valid	Valid	n/a	A new user is created and is taken to the main page
TC 8	Scenario 6- New User Addition	n/a	Invalid	Valid	Valid	Valid	Valid	Valid	n/a	Invalid Password

	Alternative flow Invalid entry									
TC 9		n/a	Valid	Invalid	Valid	Valid	Valid	Valid	n/a	Invalid Username
TC 10		n/a	Valid	Valid	Invalid	Valid	Valid	Valid	n/a	Invalid DOB
TC 11		n/a	Valid	Valid	Valid	Invalid	Valid	Valid	n/a	Invalid Phone No.
TC 12		n/a	Valid	Valid	Valid	Valid	Invalid	Valid	n/a	Invalid Email ID
TC 13		n/a	Valid	Valid	Valid	Valid	Valid	Invalid	n/a	Invalid OTP
TC 14	Scenario 7- New User Addition Alternative flow User Already Exists	n/a	Valid	Valid	Valid	Valid	Valid	n/a	n/a	User Already Exists

Assigning actual values in Test Case Matrix

Test Case ID	Scenario and Description	Input 1: User Name / Email ID	Input 2: Password	Input 3: User Name	Input 4: DOB	Input 5 : Phone No.	Input 6 : Email ID	Input 7 : OTP	Deletion confirmed	Expected Result
TC 1	Scenario 1 - User Login	abcd123@gmail.com	Shubham@123	n/a	n/a	n/a	n/a	327668	n/a	User is Logged In to the main page
TC 2	Scenario 2 - User Login Alternative Flow Invalid Entry	ABC123@gmail.com	Shubham@123	n/a	n/a	n/a	n/a	n/a	n/a	Invalid Username / Email ID
TC 3		abcd123@gmail.com	shubam@12	n/a	n/a	n/a	n/a	n/a	n/a	Invalid Password
TC 4		abcd123@gmail.com	Shubham@123	n/a	n/a	n/a	n/a	327225	n/a	Invalid OTP
TC 5	Scenario 3 - User Login Alternative Flow User Not Found	abcd123@gmail.com	Shubham@123	n/a	n/a	n/a	n/a	n/a	n/a	User does not exist
TC 6	Scenario 4 - User Login Alternative Flow User Exits	abcd123@gmail.com	Shubham@123	n/a	n/a	n/a	n/a	n/a	n/a	User exits

TC 7	Scenario 5 - New User Addition	n/a	Shubham@123	Shubham Kumar	19/06/1999	9953569302	abcd123@gmail.com	425776	n/a	A new user is created and is taken to the main page
TC 8	Scenario 6- New User Addition Alternative flow Invalid entry	n/a	shubam@12 (Requirement of a capital letter)	Shubham Kumar	19/06/1999	9953569302	abcd123@gmail.com	425776	n/a	Invalid Password
TC 9		n/a	Shubham@123	321	19/06/1999	9953569302	abcd123@gmail.com	425776	n/a	Invalid Username
TC 10		n/a	Shubham@123	321	19/06/n ot selected	9953569302	abcd123@gmail.com	425776	n/a	Invalid DOB
TC 11		n/a	Shubham@123	Shubham Kumar	19/06/1999	99535302	abcd123@gmail.com	425776	n/a	Invalid Phone No.
TC 12		n/a	Shubham@123	Shubham Kumar	19/06/1999	9953569302	ABC@g	425776	n/a	Invalid Email ID
TC 13		n/a	Shubham@123	Shubham Kumar	19/06/1999	9953569302	abcd123@gmail.com	425772	n/a	Invalid OTP
TC 14	Scenario 7- New User Addition Alternative flow User Already Exists	n/a	Shubham@123	Shubham Kumar	19/06/1999	9953569302	abcd123@gmail.com	n/a	n/a	User Already Exists