

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI-590018**



**“DBMS Laboratory with Mini project”
(Subject Code: 18CSL58)**

ON

“AIRLINE RESERVATION SYSTEM”

Submitted in partial fulfilment for 5th semester for the Award of Degree of

**BACHELOR OF ENGINEERING
IN
INFORMATION SCIENCE AND ENGINEERING**

**Ananya R(1EP19IS003)
Asha K(1EP19IS010)
Ashwaqulla Baig(1EP19IS011)
Bal Kishan Reddy(1EP19IS013)**

**UNDER THE GUIDANCE OF
Prof. INDUMATHI S
Prof. VANDANA N S
Dept. of ISE, EPCET**



**Department of Information Science and Engineering
Jnana Prabha Campus, Bidarahalli,
Bangalore – 560 049
2021 -2022**



(Affiliated to Visvesvaraya Technological University, Belagavi)
Jnana Prabha Campus, Bidarahalli,
Bangalore – 560 049

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the **DBMS Laboratory with Mini project (18CSL58)** entitled “**AIRLINE RESERVATION SYSTEM**” is a bonafide work carried out by **Ananya R,** bearing USN **1EP19IS003**, **Asha K,** bearing **1EP19IS010**, **Ashwaqulla Baig,** bearing USN **1EP19IS011**, **Bal Kishan Reddy,** bearing USN **1EP19IS013** in partial fulfilment of 5th semester for the award of, **Bachelor of Engineering in Information Science and Engineering** under **Visvesvaraya Technological University, Belagavi** during the year **2021-2022**. This report has been approved as it satisfies the academic requirements in respect of DBMS Mini Project work prescribed for the award of the said degree.

GUIDE

Prof.Indumathi S
Prof.Vandana N S

HOD

Prof.Kemparaju N
Prof & Head

PRINCIPAL

Dr.T.K. Sateesh
Principal

External examiner's

- 1)
- 2)

Date and sign

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. We would like to take this opportunity to thank them all.

First and foremost we would like to express our sincere regards and thanks to **Mr.Promod Gowda** and **Mr. Rajiv Gowda**, CEO's, East Point Group of Institutions, Bangalore, for providing necessary infrastructure and creating good environment.

We express our gratitude to **Dr.T.K. Sateesh**, Principal, EPCET who has always been a great source of inspiration.

We express our sincere regards and thanks to **Prof. Kemparaju N**, Professor and Head of Department of Information Science and Engineering, EPCET, Bangalore, for his encouragement and support.

We are grateful to acknowledge the guidance and encouragement given to us by **Prof.Indumathi S**, Assistant Professor and **Prof.Vandana N S**, Assistant Professor, Department of Information Science and Engineering, EPCET, Bangalore, as the project coordinator and guides who have rendered a valuable assistance.

We also extend our thanks to the entire faculty of the **Department of Information Science and Engineering**, EPCET, Bangalore, who have encouraged us throughout the course of the project work.

Last, but not the least, we would like to thank our family and friends for their inputs to improve the project.

Ananya R:1EP19IS003

Asha K:1EP19IS010

Ashwaqulla Baig:1EP19IS011

Bal Kishan Reddy:1EP19IS013

ABSTRACT

Airline Reservation System is a computerized system developed to store and retrieve information and conduct transactions related to air travel. The project is aimed at exposing the relevance and importance of Airline Reservation Systems. This Project is developed using HTML, PHP language, CSS for styling and MySQL for Database. It is projected towards enhancing the relationship between customers and airline agencies using Airline Reservation Systems, and thereby making it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations.

CONTENTS

| CHAPTER NO. | TOPICS | PAGE NO. |
|-------------|--|----------|
| 1 | Introduction 1.1 Problem Definition 1.2 Admin Panel 1.3 User Panel | 8 |
| 2 | Requirement specifications 2.1 Hardware configuration. 2.2 Software configuration. | 9 |
| 3 | Tool Description 3.1 Overview of Front-End Application. 3.2 Overview of Back-End Application | 10 |
| 4 | Requirement Analysis 4.1 E-R Diagram. 4.2 Schema Diagram. | 13 |
| 5 | Table Description 5.1 Database Design | 15 |
| 6 | Table with Values 6.1 Output design | 18 |
| 7 | Implementation 7.1 Sample code | 22 |
| 8 | Testing 8.1 Unit Testing 8.2 Integration Testing 8.3 User acceptance Testing 8.4 Test Cases | 29 |
| 9 | Screen Shots | 31 |
| | Conclusion | 37 |
| | References | 38 |

LIST OF FIGURES

| Figure No. | Figure Name | Page No. |
|------------|----------------------------|----------|
| 4.1 | ER Diagram | 13 |
| 4.2 | Schema Diagram | 14 |
| 5.1 | Admins table description | 15 |
| 5.2 | Cities table description | 15 |
| 5.3 | Flights table description | 16 |
| 5.4 | Users table description | 17 |
| 5.5 | Feedback table description | 17 |
| 9.1 | Home page | 31 |
| 9.2 | Search Results page | 31 |
| 9.3 | Payment's page | 32 |
| 9.4 | Feedback page | 32 |
| 9.5 | Signup page | 33 |
| 9.6 | Admin Login page | 33 |
| 9.7 | Admin Home page | 34 |
| 9.8 | Add Flight page | 34 |
| 9.9 | Delete Flight page | 35 |
| 9.10 | Update Flight page | 35 |
| 9.11 | Cancel Booking page | 36 |

LIST OF TABLES

| TABLE NO. | DESCRIPTION | PAGE No. |
|-----------|----------------------|----------|
| 1.1 | Admins Output table | 18 |
| 1.2 | Cities Output table | 19 |
| 1.3 | Flights Output table | 20 |
| 1.4 | Users Output table | 20 |
| 1.5 | Contact Output table | 21 |
| 1.6 | Test cases | 30 |

CHAPTER 1

INTRODUCTION

1.1 Problem definition

An airline reservation system (ARS) is part of the so-called passenger service systems (PSS), which are applications supporting the direct contact with the passenger ARS eventually evolved into the computer reservations system (CRS). A computer reservation system is used for the reservations of a particular airline and interfaces with a global distribution system (GDS) which supports travel agencies and other distribution channels in making reservations for most major airlines in a single system. Today all persons are busy with their schedule, and no one has time to make a trip for holidays with their family. And this Airline Reservation Process is very difficult to understand in General meaning. But we are providing a Solution for that Problem. This system provides a facility to easy access towards a customer and a real time user. They can easily connect through it and just 3 steps. There is no requirement for any type of Agent. We are giving all this facility in one project called “Airline Reservation System”.

The main objectives of this project work are:

1.2 Admin Panel:

The system helps the admin to:

- Allot different available flights to the user.
- Vacate the seats for the user.
- Manages adding, deleting and updating of flight details.

1.2 User Panel:

- Interact and search easily.
- Can also book his flight by choosing available flights.

CHAPTER 2

REQUIREMENT SPECIFICATIONS

2.1 Hardware Requirements:

| | |
|----------------------|--------------|
| Processor Brand | : Intel |
| Processor Type | : Core i5 |
| Processor Speed | : 2 GHz |
| Processor Count | : 1 |
| RAM Size | : 512 MB |
| Memory Technology | : DDR4 |
| Computer Memory Type | : DDR4 SDRAM |
| Hard Drive Size | : 500 MB |

2.2 Software Requirements:

| | |
|-----------------------|----------------------------------|
| Operating system | : Windows 10 |
| Application server | : PHP Server |
| Front end | : PHP, HTML, CSS styles |
| Browser | : Google chrome, Mozilla Firefox |
| Connectivity | : Internet |
| Database connectivity | : MySQL |

CHAPTER 3

TOOL DESCRIPTION

3.1 Overview of Front-End Application:

PHP Language:

PHP started out as a small open-source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994. PHP is a recursive acronym for "PHP: Hypertext Pre-processor". PHP is a server-side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.

It is integrated with several popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server. PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the UNIX side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time. PHP supports many major protocols such as POP3, and LDAP. PHP4 added support for Java and distributed object architectures, making n-tier development a possibility for the first time. PHP language tries to be as forgiving as possible. PHP Syntax is C Like.

Hypertext Markup Language:

Hypertext markup language (HTML) is the major markup language used to display Web pages on the Internet. In other words, Web pages are composed of HTML, which is used to display text, images or other resources through a Web browser. All HTML is plain text, meaning it is not compiled and may be read by humans. The file extension for an HTML file is .htm or .html.

HTML was never designed for the Web that exists today, as it is just a markup language with severe limitations, in terms of control and design. Numerous technologies have been used to work around this issue - the most significant being cascading style sheet (CSS).

Cascading Style Sheets:

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable. CSS handles the look and feel part of a web page. Using CSS, you can control the colour of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colours are used, as well as a variety of other effects. CSS is easy to learn and understand but it provides a powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

3.2 Overview of Back-End Application:

MySQL:

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds. Other kinds of data stores can also be used, such as files on the file system or large hash tables in memory, but data fetching and writing would not be so fast and easy with those types of systems.

Nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as Foreign Keys.

A Relational Database Management System (RDBMS) is software that:

- Enables you to implement a database with tables, columns and indexes.
- Guarantees the Referential Integrity between rows of various tables.
- Updates the indexes automatically.
- Interprets an SQL Query and combines information from various tables.

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is released

under an open-source license. So, you have nothing to pay to use it. MySQL is a very powerful program. It handles a large subset of the functionality of the most expensive and powerful database packages. MySQL uses a standard form of the well-known SQL data language. MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc. MySQL works very quickly and works well even with large data sets. MySQL is very friendly to PHP, the most appreciated language for web development.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 E-R diagram

ER Diagram: ER Diagram is a high-level conceptual data model diagram. Entity-Relation model is based on the notion of real-world entities and the relationship between them. ER modelling helps you to analyse data requirements systematically to produce a well-designed database.

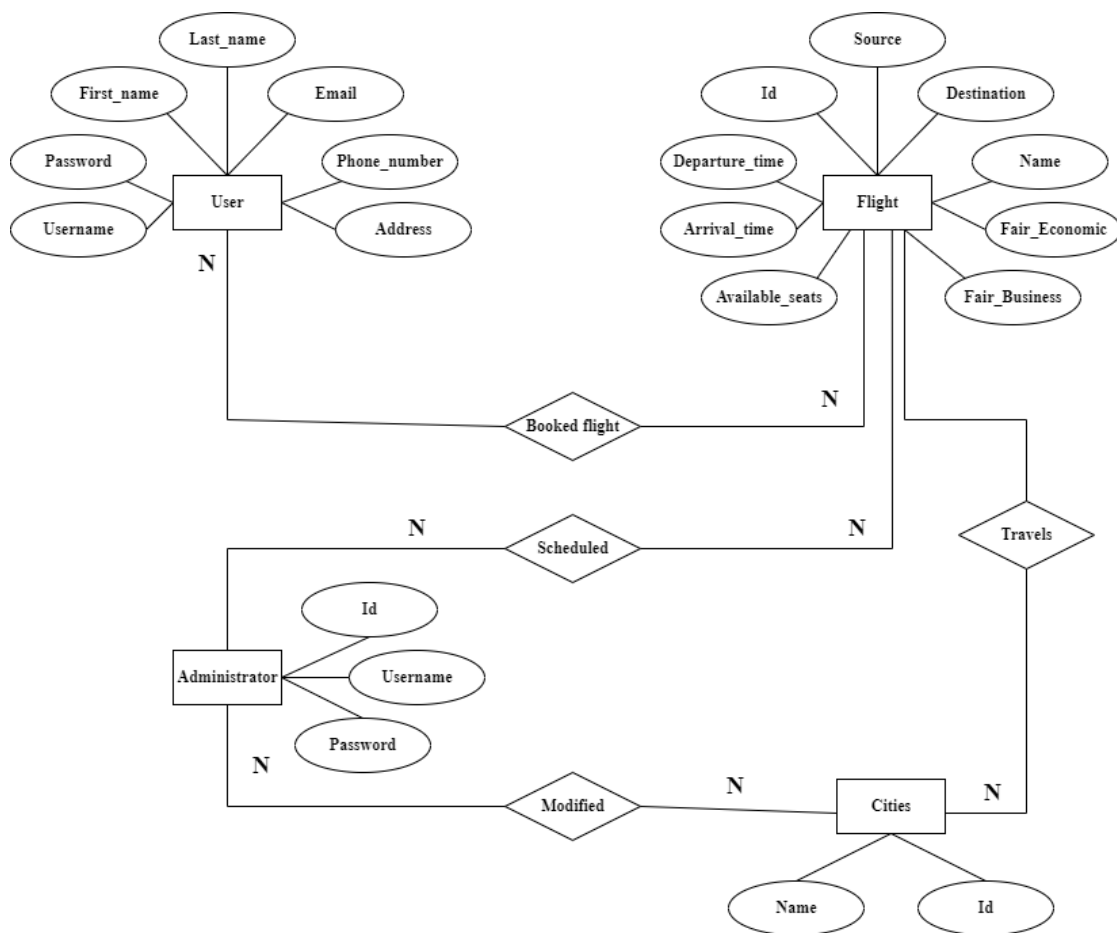


Fig. 4.1 ER diagram of Airline Reservation System.

4.2 SCHEMA DIAGRAM

Schema diagram: A schema diagram is the skeleton structure that represents the logical view of the entire database. It contains a descriptive detail of the database.

The Fig 4.2 shows the schema diagram for **AIRLINE RESERVATION SYSTEM**.

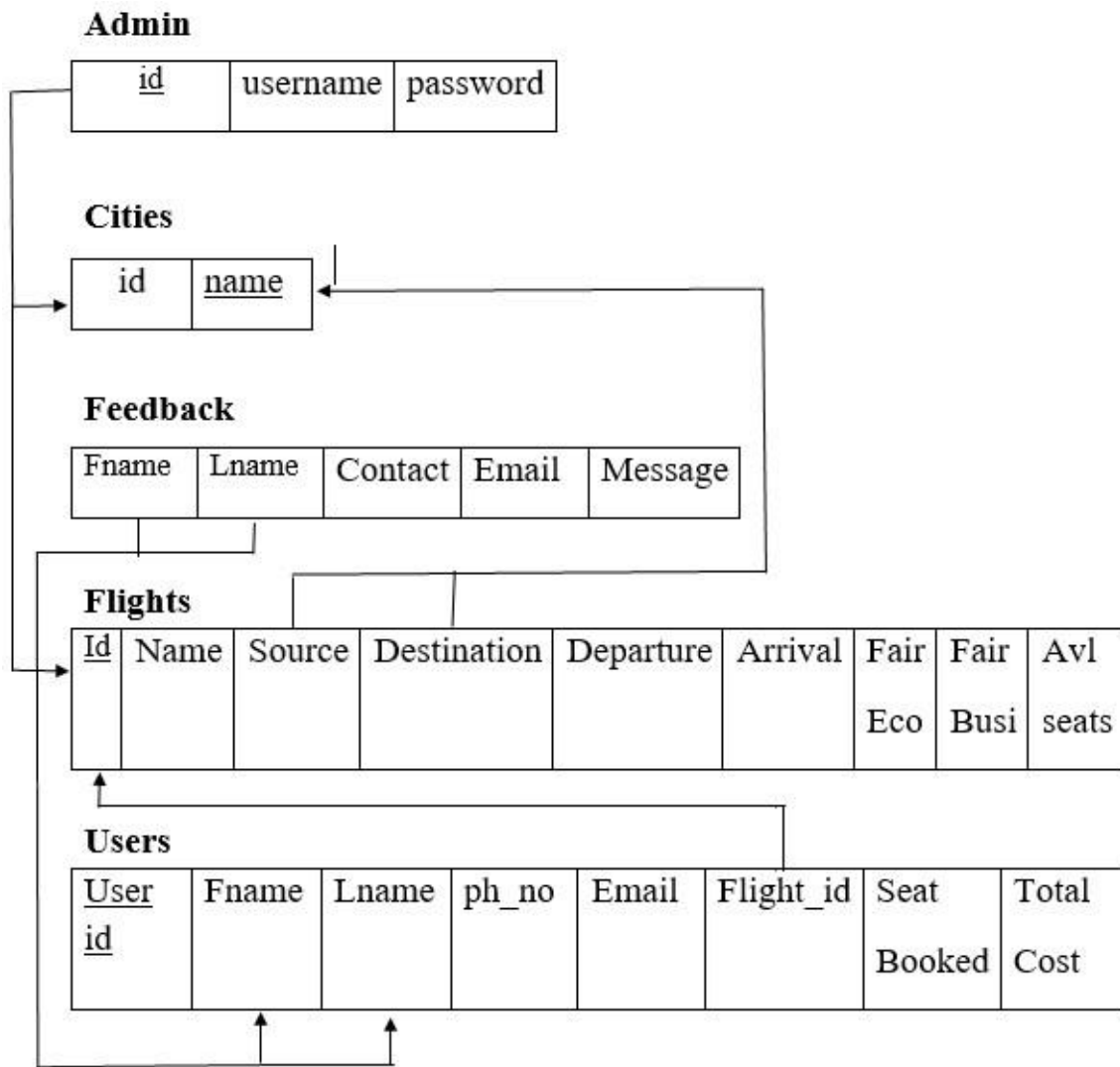


Figure 4.2 Schema diagram for Airline Reservation system

CHAPTER 5

TABLE DESCRIPTION

5.1 Database Design

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MS Access database has been chosen for developing the relevant databases.

The following are the tables that are involved in the proposed system:

Admins table: Admins table is used to store login details of registered admins. It consists of three attributes which are Id, Username, Password. Here id is used as primary key.

desc admins ;

| Name | Null? | Type |
|----------|----------|--------------|
| ----- | ----- | ----- |
| Id | NOT NULL | INT (10) |
| Username | NOT NULL | VARCHAR2(20) |
| Password | NOT NULL | VARCHAR2(10) |

Figure 5.1 Admins table description

Cities table: Cities table contains list of added cities. It includes only one attribute which is Name. Here Name is used as primary key.

desc cities;

| Name | Null? | Type |
|-------|----------|--------------|
| ----- | ----- | ----- |
| Name | NOT NULL | VARCHAR (15) |

Figure 5.2 Cities table description

Flights table: Flights table is used to store details of the running flights which contains Id, Name, Source, Destination, Departure, Arrival, Fair_Economic, Fair_Bussiness, Available_seats. Here Id is used as primary key.

desc flights;

| Name | Null? | Type |
|-----------------|----------|-----------|
| ----- | ----- | ----- |
| Id | NOT NULL | INT (11) |
| Name | NOT NULL | TEXT |
| Source | NOT NULL | TEXT |
| Destination | NOT NULL | TEXT |
| Departure | NOT NULL | DATE |
| Arrival | NOT NULL | DATE |
| Fair_Economic | NOT NULL | INT (11) |
| Fair_Bussiness | NOT NULL | INT (11) |
| Available_seats | NOT NULL | INT (100) |

Figure 5.3 Flights table description

Users table: Users table stores the details of the users registered such as UserId, FirstName, LastName, MobileNo, Email, Flight_Id, Seats_booked, Total_Cost. Here UserId is the primary key.

desc users;

| Name | Null? | Type |
|--------------|----------|--------------|
| ----- | ----- | ----- |
| UserId | NOT NULL | INT (4) |
| FirstName | NOT NULL | TEXT |
| LastName | NOT NULL | TEXT |
| MobileNo | NOT NULL | BIGINT (10) |
| Email | NOT NULL | VARCHAR (50) |
| Flight_Id | NOT NULL | INT (11) |
| Seats_booked | NOT NULL | INT (11) |
| Total_Cost | NOT NULL | INT (11) |

Figure 5.4 Users table description

Feedback table: Feedback table is used to store the details of each user who provides feedback. It contains Name, Contact, Email, Message.

desc feedback;

| Name | Null? | Type |
|---------|----------|--------------|
| ----- | ----- | ----- |
| Name | NOT NULL | TEXT |
| Contact | NOT NULL | BIGINT (10) |
| Email | NOT NULL | VARCHAR (20) |
| Message | NOT NULL | TEXT |

Figure 5.5 Feedback table description

CHAPTER 6

TABLE WITH VALUES

6.1 Output Design

Designing computer output should proceed in an organized, well throughout manner; the right output element is designed so that people will find the system whether or executed. When we design an output, we must identify the specific output that is needed to meet the system. The usefulness of the new system is evaluated based on their output.

Once the output requirements are determined, the system designer can decide what to include in the system and how to structure it so that they require output can be produced. For the proposed software, it is necessary that the output reports be compatible in format with the existing reports. The output must be concerned to the overall performance and the system's working, as it should. It consists of developing specifications and procedures for data preparation, those steps necessary to put the inputs and the desired output, ie maximum user friendly. Proper messages and appropriate directions can control errors committed by users. The output design is the key to the success of any system. Output is the key between the user and the sensor. The output must be concerned to the system's working, as it should. Output design consists of displaying specifications and procedures as data presentation. User never left with the confusion as to what is happening without appropriate error and acknowledges message being received. Even an unknown person can operate the system without knowing anything about the system.

Admins Output table: Admin Output table is used to store login credentials of admins which later can be used to login and modify or view the details of flights.

SQL>> select * from admins;

| USERID | USERNAME | PASSWORD | USERTYPE |
|--------|-----------|-------------|----------|
| ----- | ----- | ----- | ----- |
| 100 | root | root | user |
| 104 | fleetaway | fleetaway@7 | admin |
| 105 | bkr | bkr@7 | admin |

Table 1.1 Admins Output table

Cities Output table: Cities Output table contains the list of Cities which provides user to select the desired city for flight booking.

SQL> select * from cities;

| NAME |
|--------------|
| ----- |
| Bangalore |
| Chennai |
| Delhi |
| Dubai |
| Goa |
| Iceland |
| Italy |
| Kolkata |
| Kullu Manali |
| London |
| Maldives |
| Miami |
| Mumbai |
| Nepal |
| New York |
| Switzerland |

Table 1.2 Cities Output table

Flights Output table: Flights Output table is used to store the flight details regarding flight name, source , destination and fare.

SQL> select * from flights;

| Id | Name | Source | Destination | Departure | Arrival | Fair_Economic | Fair_Bussiness | Available_Seats |
|------|----------|-----------|-------------|------------|------------|---------------|----------------|-----------------|
| --- | ----- | ----- | ----- | ----- | ----- | ----- | ----- | ----- |
| 101 | Emirates | Bangalore | Dubai | 2022-03-26 | 2022-04-01 | 12000 | 24000 | 44 |
| 102 | Emirates | Delhi | Italy | 2022-03-26 | 2022-04-25 | 25000 | 50000 | 22 |
| 103 | SpiceJet | Bangalore | Delhi | 2022-03-26 | 2022-04-08 | 4000 | 8000 | 60 |
| 104 | GoAir | Bangalore | Chennai | 2022-03-26 | 2022-03-27 | 3500 | 7000 | 50 |
| 105 | Indigo | Delhi | Bangalore | 2022-03-26 | 2022-02-30 | 3000 | 5500 | 48 |
| 1000 | Indigo | Bangalore | Delhi | 2022-03-26 | 2022-02-30 | 3500 | 6000 | 43 |
| 1001 | Emirates | Bangalore | Switzerland | 2022-03-26 | 2022-05-07 | 30000 | 55000 | 25 |

Table 1.3 Flights Output table

Users Output table: Users Output table is used to store the details of users flight details which can be identified by his user id and flight id.

SQL> select * from users;

| UserId | FirstName | LastName | MobileNo | Email | Flight_Id | Seats_booked | Total_Cost |
|--------|------------|----------|------------|--|-----------|--------------|------------|
| ----- | ----- | ----- | ----- | ----- | ----- | ----- | ----- |
| 110 | Ashwaqulla | Baig | 6985214554 | ashwaq011@gmail.com | 101 | 1 | 48000 |
| 119 | Asha | K | 7584444369 | asha010@gmail.com | 500 | 4 | 14000 |
| 200 | Bal Kishan | Reddy | 6525241894 | bkrreddy07@gmail.com | 102 | 1 | 25000 |
| 203 | Ananya | R | 9009788542 | ananya003@gmail.com | 2120 | 2 | 220000 |

Table 1.4 Users Output table

Contact Output table: Contact Output table is used to store the details of the users who have given the feedback.

SQL> select * from contact;

| Name | Contact | Email | Message |
|------------------|------------|--|--------------------------|
| ----- | ----- | ----- | ----- |
| Ashwaqulla Baig | 6985214554 | ashwaq011@gmail.com | Helpful & User-friendly! |
| Asha K | 7584444369 | asha010@gmail.com | Enjoyed! |
| Ananya | 9009788542 | ananya003@gmail.com | Best Policy! |
| Bal Kishan Reddy | 6525241894 | bkrredy07@gmail.com | Flamboyant! |

Table 1.5 Contact Output table

CHAPTER 7

IMPLEMENTATION

7.1 Sample code

Code to establish connection with the database:

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "database_file";

// Create connection
$conn = mysqli_connect($servername, $username, $password,$dbname);

// Check connection if (!$conn) {    die("Connection failed: " .
mysqli_connect_error());
}

?>
```

Code to perform login page:

```
<?php

include 'config.php';

if($_SERVER["REQUEST_METHOD"] == "POST") {
    // username and password sent from form
```

```
$username = $_POST['username'];
$password = $_POST['password'];

$sql = "SELECT Id FROM admins WHERE Username = '$username' and Password = '$password'";
$result = mysqli_query($conn,$sql);
$row = mysqli_fetch_array($result,MYSQLI_ASSOC);
//$active = $row['active'];

$count = mysqli_num_rows($result);

// If result matched $myusername and $mypassword, table row must be 1 row

if($count == 1) {
    $_SESSION["myusername"]=$username;
    $_SESSION['login_user'] = $username;

    header("location: welcome.php");
}else {
    $error = "Your Login Name or Password is invalid<br><br>";
    echo"<link href=\"https://cdn.jsdelivr.net/npm/bootstrap@5.0.0beta1/dist/css/bootstrap.min.css\"
rel=\"stylesheet\"
integrity=\"sha384giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHYTfAAsrekWkmP1\
\"
crossorigin=\"anonymous\"><div class=\"alert alert-success\" role=\"alert\"><h4 class=\"alert-
heading\">Sorry!</h4><p>$error</p><hr><a href=\"admin.html\"
class=\"button\">Login</a></div>";
    }
}

mysqli_close($conn);
?>
```

Code to perform insertion:

```
<?php
include 'config.php';

$flight_id = $_POST['flight_id'];
$total_cost = $_POST['price'];
$total_passengers = $_POST['total_passengers'];
$firstname = $_POST['firstname'];
$lastname = $_POST['lastname'];
$mob_no = $_POST['mob_no'];
$email = $_POST['email'];

$sql = "INSERT INTO users (FirstName,LastName, MobileNo, Email, Flight_Id,
Seats_booked,Total_Cost)
VALUES('$firstname','$lastname','$mob_no','$email','$flight_id','$total_passengers','$total_c ost')";

$result = mysqli_query($conn,$sql);

$sql = "SELECT Available_seats FROM flights WHERE Id =$flight_id";
$result = mysqli_query($conn,$sql);

$row = mysqli_fetch_assoc($result);
$updated_seats = $row['Available_seats'] - $total_passengers;

$sql = "UPDATE flights SET Available_seats= $updated_seats WHERE Id =$flight_id";
$result = mysqli_query($conn,$sql); mysqli_close($conn);

?>
```


Code to perform search result:

```
<?php
include 'config.php'; /*function
test_input($data) {
    $data = trim($data);
    $data = stripslashes($data); $data =
htmlspecialchars($data); return $data;
} */
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $source = $_POST["source"];
    $destination = $_POST["destination"];
    $departure = $_POST["departdate"];
    $trip = $_POST["trip"]; if ($trip ==
'return') {
        $arrival = $_POST["arrivedate"];
    }
    $adults = $_POST["adults"];
    $trip_class = $_POST['travel_class'];
    $total_passengers = $adults;

} if($trip == 'oneway'){
$sql = "SELECT * FROM flights WHERE Source = '$source' AND Destination = '$destination'
AND '$departure'>=Departure AND Available_seats>0 ";
$result = mysqli_query($conn,$sql);
} else {
    $sql = "SELECT * FROM flights WHERE Source = '$source' AND Destination = '$destination' AND
'$departure'>=Departure AND '$arrival'<=Arrival AND Available_seats>0
";
    $result = mysqli_query($conn,$sql);
```

```
} echo"<table border = '1'>"; echo
"<tr><th>Id</th><th>Name</th><th>Source</th><th>Destination</th><th>Fare</th><th>A
ction</th></tr>"; if ($Strip_class == 'economic') { if ($Strip == 'oneway') { while ($row =
mysqli_fetch_assoc($result)) {
    $price = $row['Fair_Economic']*$adults;
    $id = $row['Id']; echo
"<tr><td>{$row['Id']}</td><td>{$row['Name']}</td><td>{$row['Source']}</td><td>{$row['
Destination']}</td><td>{$price}</td><td><form id= \"Passing\" method=\"post\"
action=\"book_flight.php\">
<input name=\"Id\" type=\"hidden\" value=\"$id\">
<input name=\"price\" type=\"hidden\" value=\"$price\">
<input name=\"total_passengers\" type=\"hidden\" value=\"$total_passengers\">
<input name=\"submit\" type=\"submit\" value=\"Book\">
</form></td></tr>";
    }
}
else {
    while ($row = mysqli_fetch_assoc($result)) { $price_temp =
$row['Fair_Economic']*$adults;
    $price = $price_temp*2; $id =
$row['Id']; echo
"<tr><td>{$row['Id']}</td><td>{$row['Name']}</td><td>{$row['Source']}</td><td>{$row['
Destination']}</td><td>{$price}</td><td><form id= \"Passing\" method=\"post\"
action=\"book_flight.php\">
<input name=\"Id\" type=\"hidden\" value=\"$id\">
<input name=\"price\" type=\"hidden\" value=\"$price\">
<input name=\"total_passengers\" type=\"hidden\" value=\"$total_passengers\">
<input name=\"submit\" type=\"submit\" value=\"Book\">
</form></td></tr>";
    }
}
```

```
} } else { if ($strip == 'oneway') { while ($row =  
mysqli_fetch_assoc($result)) {  
    $price = $row['Fair_Business']*$adults;  
    $id = $row['Id']; echo  
"<tr><td>{$row['Id']}</td><td>{$row['Name']}</td><td>{$row['Source']}</td><td>{$row['  
Destination']}</td><td>{$price}</td><td><form id=    \"Passing\"    method=\"post\"  
action=\"book_flight.php\">  
    <input name=\"Id\" type=\"hidden\" value=\"$id\">  
    <input name=\"price\" type=\"hidden\" value=\"$price\">  
    <input name=\"total_passengers\" type=\"hidden\" value=\"$total_passengers\">  
    <input name=\"submit\" type=\"submit\" value=\"Book\">  
    </form></td></tr>";  
    } } else {  
while ($row =  
mysqli_fetch_asso  
c($result)) {  
    $price_temp = $row['Fair_Business']*$adults+0.5*$row['Fair_Business']*$childrens;  
    $price = $price_temp*2;    $id =  
$row['Id']; echo  
"<tr><td>{$row['Id']}</td><td>{$row['Name']}</td><td>{$row['Source']}</td><td>{$row['  
Destination']}</td><td>{$price}</td><td><form id=    \"Passing\"    method=\"post\"  
action=\"book_flight.php\">  
    <input name=\"Id\" type=\"hidden\" value=\"$id\">  
    <input name=\"price\" type=\"hidden\" value=\"$price\">  
    <input name=\"total_passengers\" type=\"hidden\" value=\"$total_passengers\">  
    <input name=\"submit\" type=\"submit\" value=\"Book\">  
    </form></td></tr>";  
    }  
} } echo "</table>";  
mysqli_close($conn);
```

?>

Code to perform logout:

```
<?php
    session_start();  if(session_destroy()) {
header("Location: website.php");
    }
?>
```

CHAPTER 8

TESTING

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is the process of executing the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met, and the user requirements are satisfied. The aim is quality assurance.

8.1 Unit Testing

The software units in a system are modules and routines that are assembled and integrated to perform a specific function. Unit testing focuses first on modules, independently of one another, to locate errors. This enables, to detect errors in coding and logic that are contained within each module. This testing includes entering data and ascertaining if the value matches to the type and size supported by java. The various controls are tested to ensure that each performs its action as required.

8.2 Integration Testing

Data can be lost across any interface, one module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integration testing is a systematic testing to discover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here the Server module and Client module options are integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data.

8.3 User Acceptance Testing

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the system users at time of developing and making changes whenever required.

8.4 TEST CASES

Table 1.6: Test Cases

| Test no | Test name | Input | Actual output | Expected output | Status |
|---------|-------------------------|--|-------------------------------------|-------------------------------------|--------|
| 1 | Login | Username and Password | User is successfully Authenticated. | User is successfully Authenticated. | Pass |
| 2 | Login | Wrong username and password | Invalid username or password. | Invalid username or password. | Pass |
| 3 | User Search and booking | Fill the flight details, fill the form and click on book | Users registration is successful. | Users registration is successful. | Pass |
| 4 | Add flight | Enter flight id and click on add flight | Flight is added successfully. | Flight is added successfully. | Pass |
| 5 | Delete flight | Enter flight id and click on delete flight | Flight is deleted successfully. | Flight is deleted successfully. | Pass |
| 6 | Update flight | Enter flight id and click on update flight | Flight id is updated successfully. | Flight id is updated successfully | Pass |
| 7 | Cancel flight | Enter user id and click on cancel flight | Flight booking is cancelled. | Flight booking is cancelled. | Pass |
| 8 | Logout | Click on logout | Logout successful. | Logout successful. | Pass |

CHAPTER 9

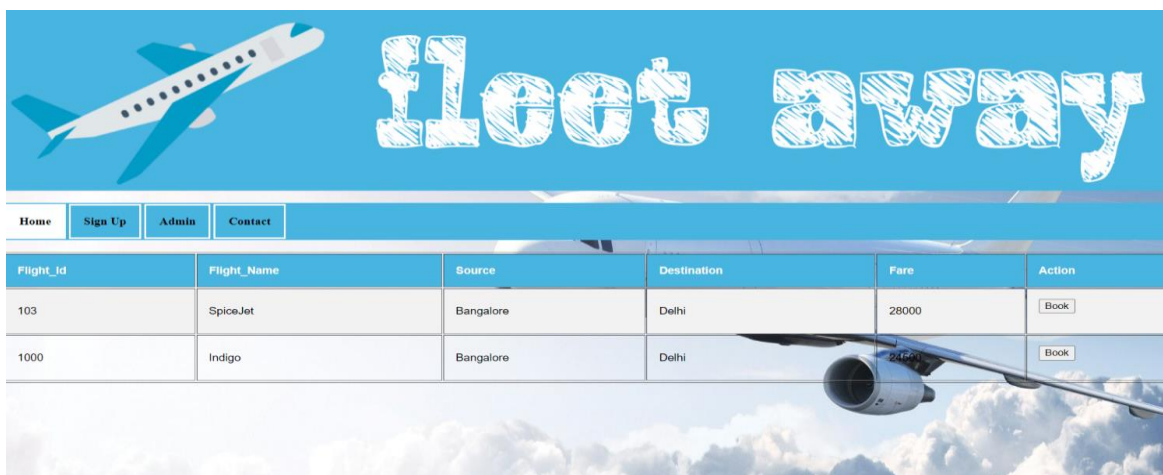
SNAPSHOTS

1.Home page: This page represents the first thing about our website. Here a user can fill out or choose the details for the following flight trip from the desired Source to Destination.



Figure 9.1: Home page

2. Search Results page: After filling out details for the flight, it takes you to the Search results page where you can see the details of flights available with an option to Book.



| Flight_Id | Flight_Name | Source | Destination | Fare | Action |
|-----------|-------------|-----------|-------------|-------|----------------------|
| 103 | SpiceJet | Bangalore | Delhi | 28000 | Book |
| 1000 | Indigo | Bangalore | Delhi | | Book |

Figure 9.2 Search Results page

3. Payment's page: This page represents the payment details for a user by entering his personal details and making a payment for the flight which is booked.

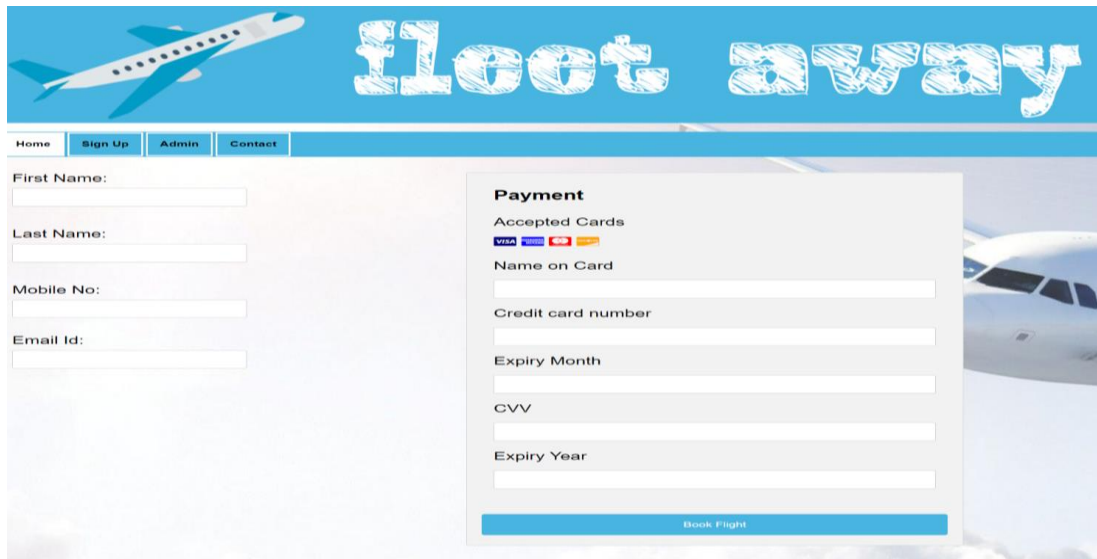


Figure 9.3 Payment's page

4.Feedback page: After Successfully booking the flight ticket, user can choose to provide feedback on the services of the company, that how the user liked the experience throughout booking of a flight ticket, if user has experienced any problem, user can write a query directly to the company.

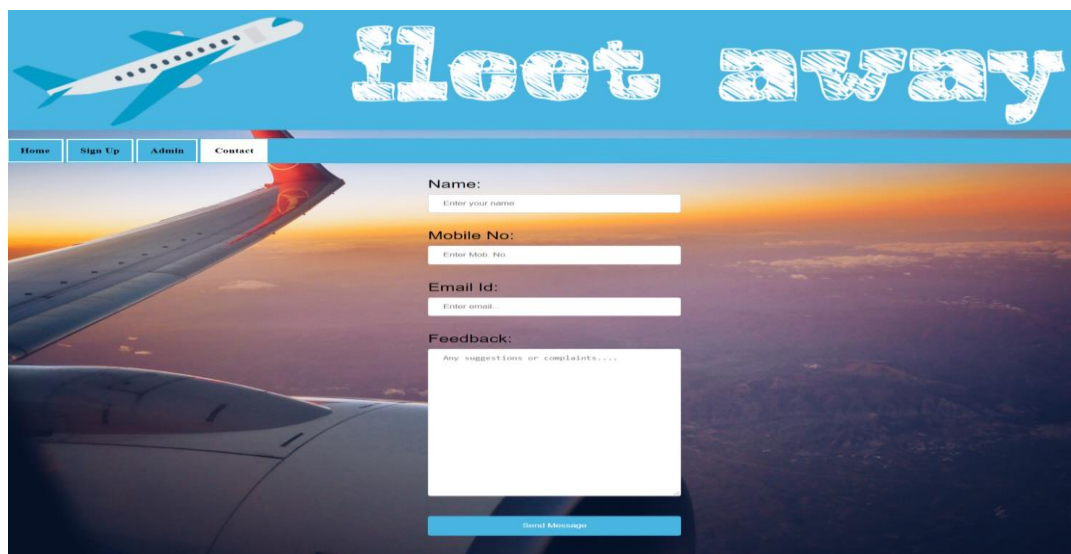


Figure 9.4 Feedback page

5.Signup page: This page provides the user credentials to sign up for the account.

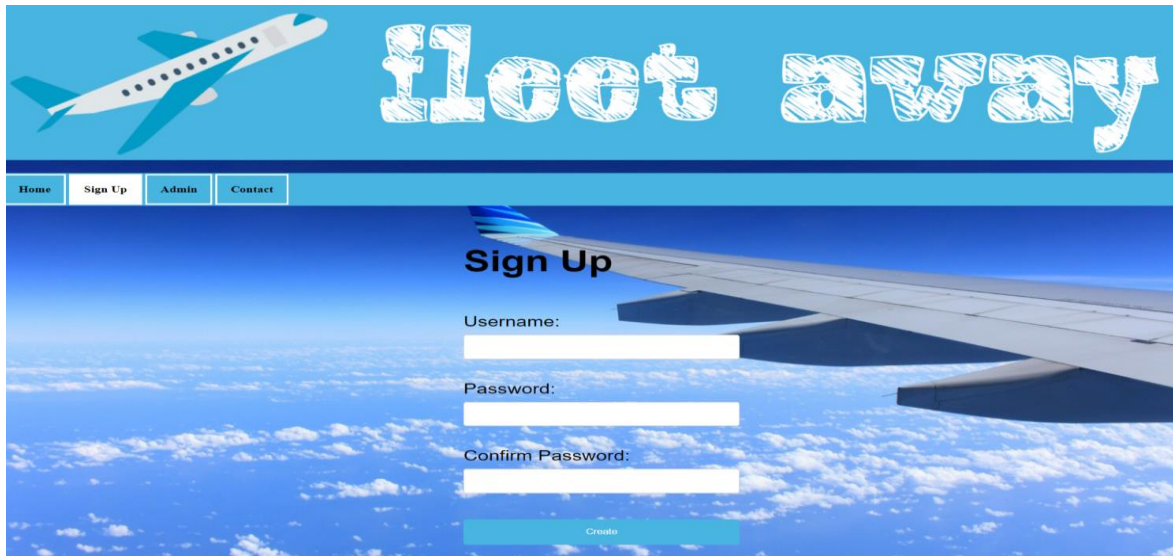


Figure 9.5 Signup page

6.Admin Login page: This page has the login credentials for admins who have access to managing the flight tickets fare, timings and place etc.



Figure 9.6 Admin Login page

7.Admin Home page: This page provides admins access to all details of a user regarding flight Source and Destination, Departure and Arrival, Amount etc.

It also allows admins to add, delete, update and modify flight details.

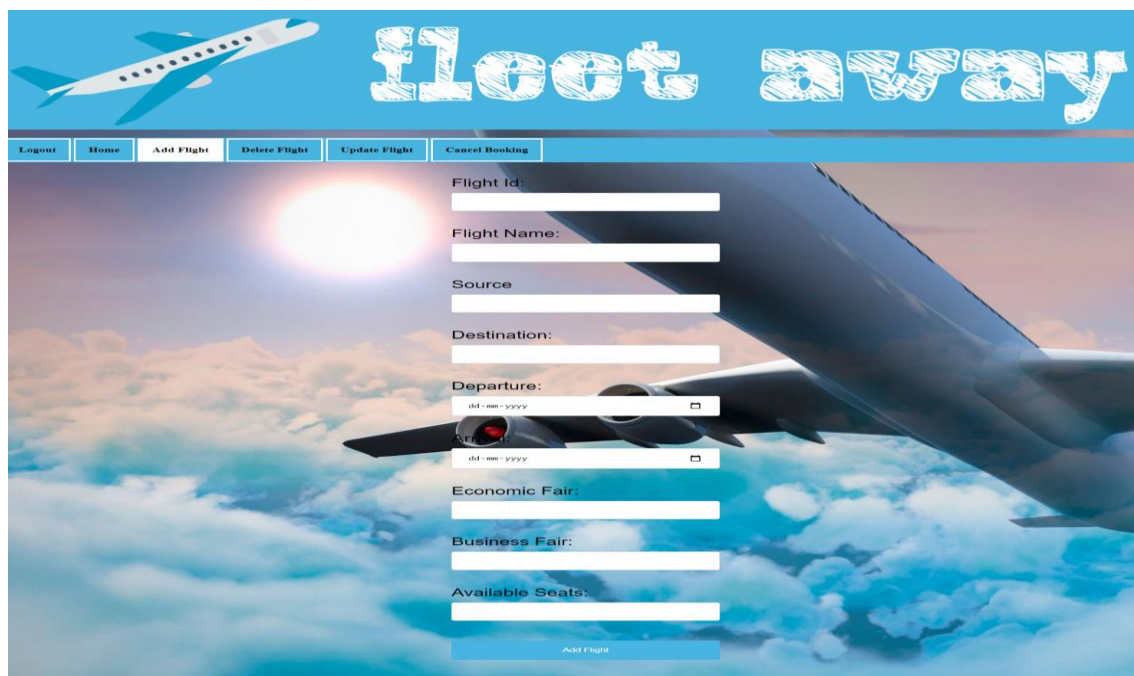


| Flight_Id | Flight_Name | Source | Destination | Departure | Arrival | Fair_Economic | Fair_Business | Available_seats |
|-----------|-------------|-----------|-------------|------------|------------|---------------|---------------|-----------------|
| 101 | Emirates | Bangalore | Dubai | 2022-03-26 | 2022-04-01 | 12000 | 24000 | 44 |
| 102 | Emirates | Delhi | Italy | 2022-03-26 | 2022-04-25 | 25000 | 50000 | 21 |
| 107 | SpiceJet | Bangalore | Delhi | 2022-03-26 | 2022-04-08 | 4000 | 8000 | 60 |
| 500 | GoAir | Bangalore | Chennai | 2022-03-26 | 2022-03-27 | 3500 | 7000 | 50 |
| 786 | Indigo | Delhi | Bangalore | 2022-03-26 | 2022-03-30 | 3000 | 5500 | 48 |
| 1001 | Indigo | Bangalore | Delhi | 2022-03-26 | 2022-03-30 | 3500 | 6000 | 43 |
| 2120 | Emirates | Bangalore | Switzerland | 2022-03-26 | 2022-05-07 | 30000 | 55000 | 23 |

| User_Id | FirstName | LastName | MobileNo | Email | Flight_Id | Seats_booked | Total_Cost |
|---------|------------|----------|------------|----------------------|-----------|--------------|------------|
| 110 | Ashwaqulla | Baig | 6985214554 | ashwaq011@gmail.com | 101 | 1 | 48000 |
| 199 | Asha | K | 7584444369 | asha010@gmail.com | 500 | 4 | 14000 |
| 200 | Bal Kishan | Reddy | 6525241894 | bkrreddy07@gmail.com | 102 | 1 | 25000 |
| 203 | Ananya | R | 9009788542 | ananya003@gmail.com | 2120 | 2 | 220000 |

Figure 9.7 Admin Home page

8.Add Flight page: This page allows admins to add the flight details which later reflects in Admin Home page.



Flight Id:

Flight Name:

Source:

Destination:

Departure:

Economic Fair:

Business Fair:

Available Seats:

Figure 9.8 Add Flight page

9.Delete Flight page: This page allows admins to delete the flight details which later reflects in Admin Home page.

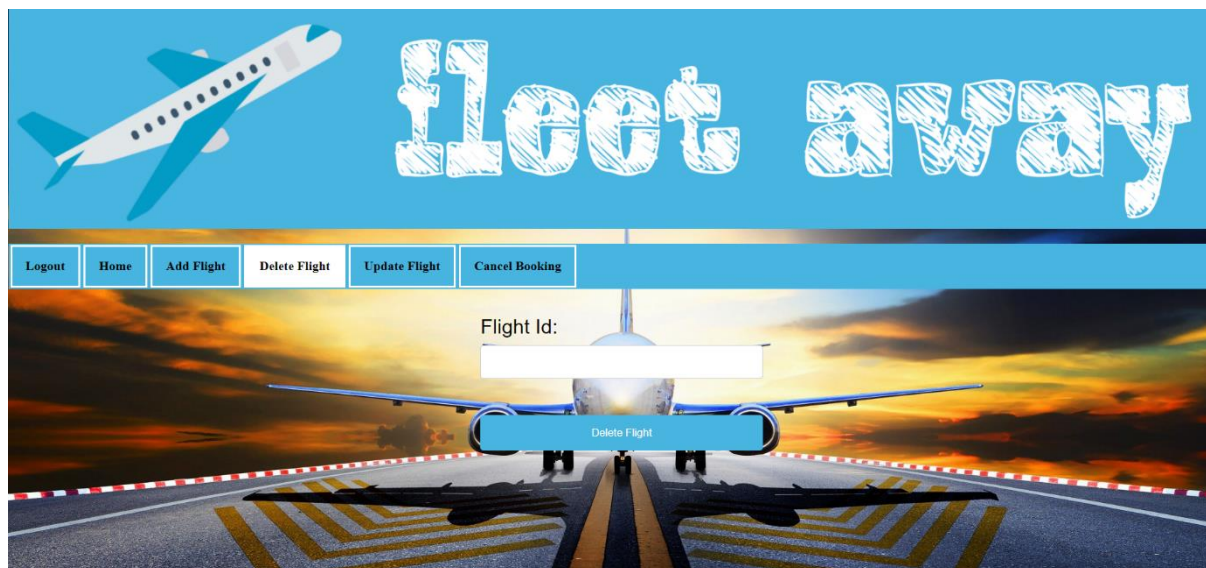


Figure 9.9 Delete Flight page

10.Update Flight page: This page allows admins to update the flight details which later reflects in Admin Home page.

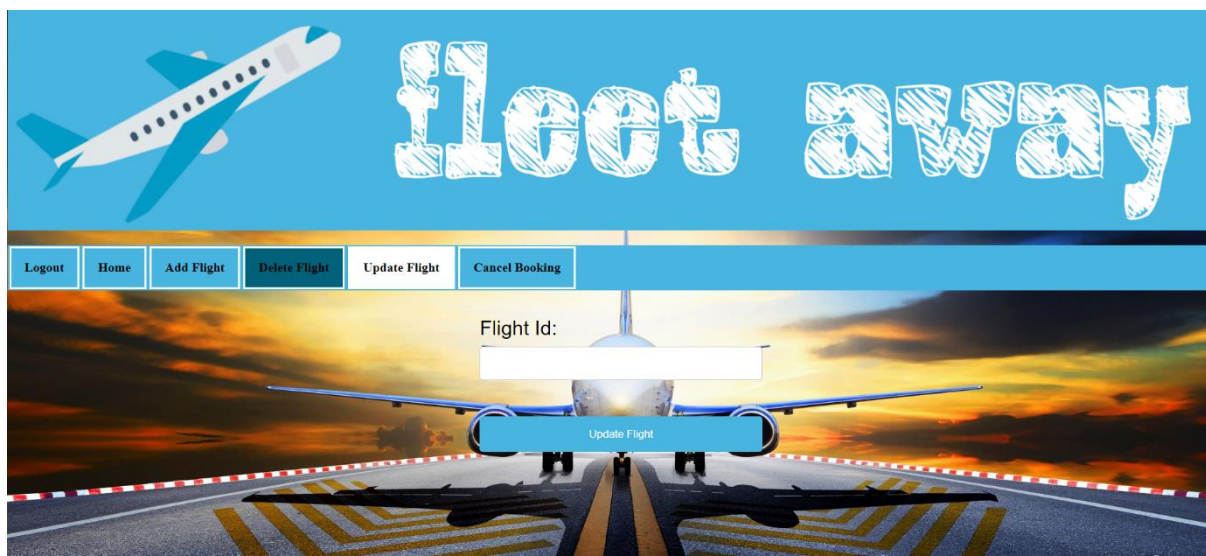


Figure 9.10 Update Flight page

11.Cancel Booking page: This page allows admins to cancel the flight booking of a user which later reflects in Admin Home page.

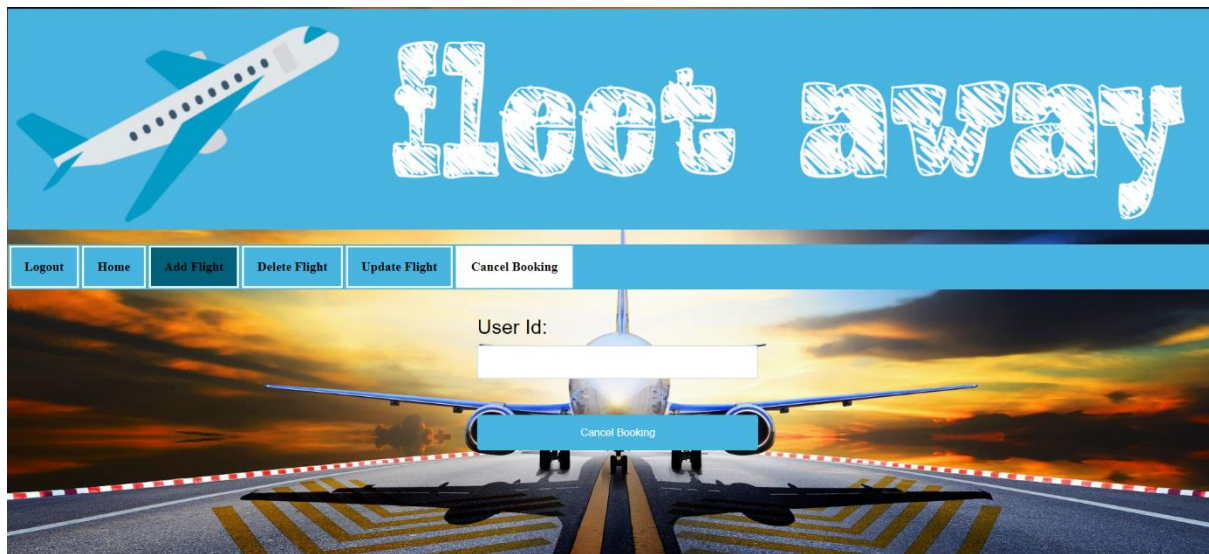


Figure 9.11 Cancel Booking page

CONCLUSION

The following Conclusion can be deduced from the project. This project offers user to enter the data through simple and interactive manner. User is provided the option of only view the records he entered earlier. Data storage and retrieval will become faster and easier to maintain. This project is a very flexible software, and it can be upgraded according to the individual trip needs. It is a user friendly and customized software for Airline Reservation admin. It has been developed to manage and automate the overall processing of any large flight booking.

Advantages:

- . High security.
- . Data redundancy can be avoided.
- . Easy to handle, update and keep record.
- . Backup data can easily generate.
- . Easy to handle.
- . Easy data updating.

REFERENCES

BOOKS:

- [1] Silberschatz Korth and Sudharshan: Database Sytems Concepts, 6th Edition, Mc-GrawHill, 2013 and
- [2] Fundatmentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.

WEBSITES:

W3Schools.com

- 1) <https://www.w3schools.com/php/>
- 2) <https://www.w3schools.com/html/>
- 3) <https://www.w3schools.com/css/default.asp>
- 4) https://www.w3schools.com/sql/sql_intro.asp

