Velammal College of Engineering and Technology,
Madurai – 625 009.(An Autonomous Institution)
Department of Computer Science and Engineering
Academic Year: 2022 - 2023 (EVEN Semester)
B.E CSE - II YEAR - IV SEMESTER

21CS210 – Database Management Systems

LaboratoryMini Project

AIRLINE RESERVATION SYSTEM

SUBMITTED BY,
Team member1: Roll No: 21CSE118 Name: C.B.Ramkishan
Team member2: Roll No: 21CSE114 Name: E.Nihil Sankar

Mark Splitup

S.No.	Criteria	Maximum Marks	Marks Obtained
1.	Application Complexity	05	
2.	Database Design	20	
3.	Frontend Design	15	
4.	Presentation	05	
5.	On time submission	05	
Total		**50**	

Course Skilled
Dr. S. Sasikala
Associate Professor, CSE

Course In-charge
Dr. A.M.Rajeswari
Associate Professor, CSE

1.Introduction

Airline reservation systems (**ARS**) are systems that allow an airline to sell their inventory (seats). It contains information on schedules and fares and contains a database of reservations (or passenger name records) and of tickets issued (if applicable). ARSs are part of 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.

2.Application Users

As an application user for an airline reservation system, your main goal is to easily and efficiently book flights, manage your reservations, and access important information related to your travel plans.

3.Functions Overview

The following functionality will be supported by the airline reservation system.

1. Flight booking .
2. Seat availabiity .
3. User accounts.

4.Detailed Functional Requirements

1.Flight search: The system should allow you to search for available flights based on your preferred dates, airports, and other parameters such as the number of passengers and class of travel.

2.Flight booking: Once you have found a flight that meets your requirements, the system should allow you to book it easily, either through a simple online booking process or by calling a customer service representative.

3.Reservation management: The system should provide you with an easy way to manage your reservations, such as the ability to view, modify, or cancel existing reservations.

4.Flight status updates: The system should provide you with up-to-date information on flight schedules, delays, cancellations, and gate changes, so you can plan accordingly.

5.Boarding passes: The system should allow you to easily access and print boarding passes, or to save them to your mobile device for convenient access.

6.Baggage tracking: The system should provide you with the ability to track your baggage from check-in to arrival, so you can be sure it arrives with you at your destination.

7.Loyalty program integration: If you are a member of the airline's loyalty program, the system should allow you to access your account information, view your rewards balance, and redeem rewards.

8.Customer support: The system should provide you with easy access to customer support, whether through a chat feature, email, or phone.

Overall, an airline reservation system should provide you with a seamless and hassle-free experience from start to finish, making it easy to book and manage your travel plans.

5.Software and Hardware Requirements

- Frontend tools : VS studio
- Service-side programming : PHP
- Backend : Mysql

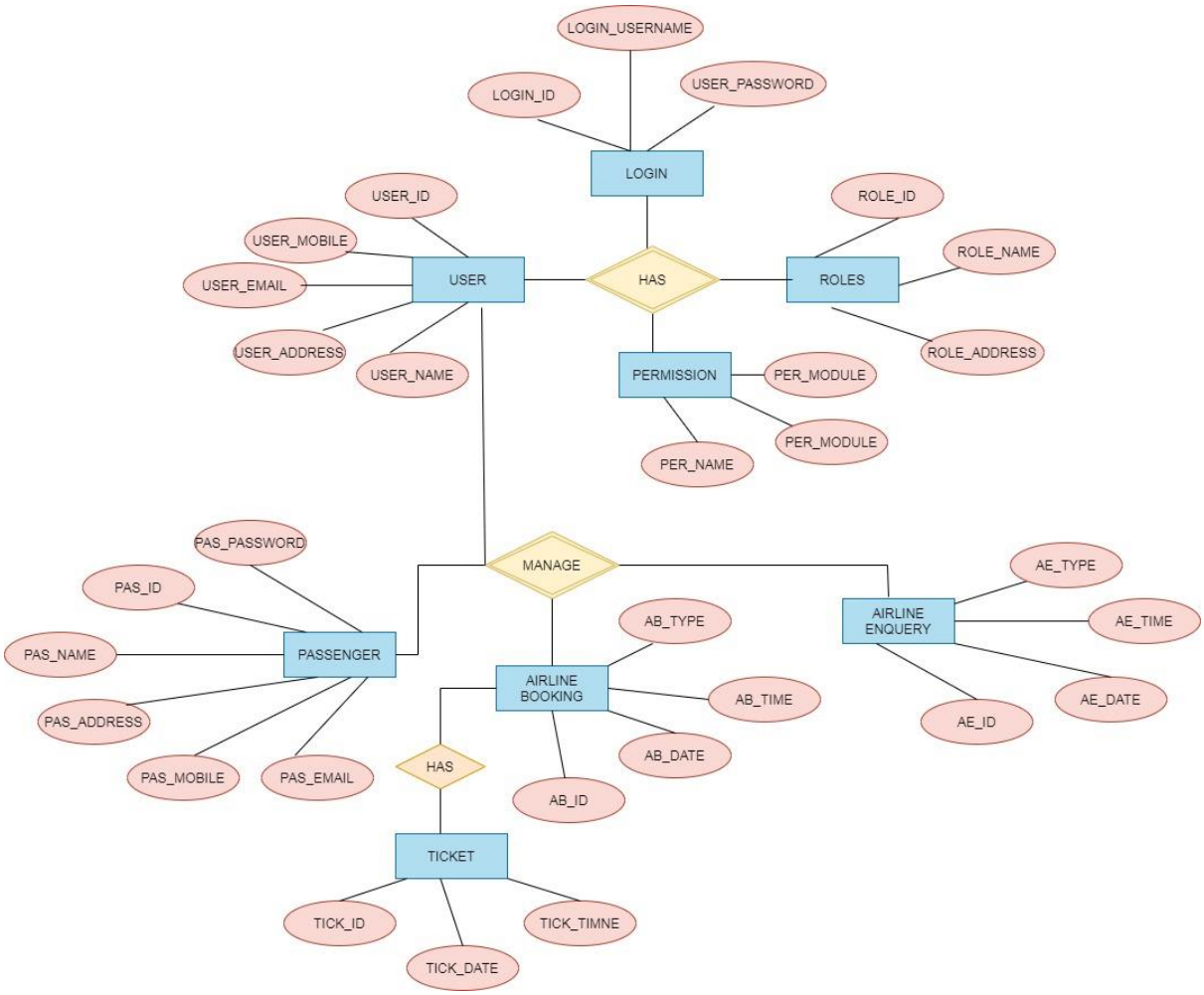
6.Performance criteria

- Capacity requirments : 10-15 Users.
- Database size : 20-30 tuples.

7.Entities and relationships

ENTITIES	RELATIONSHIP IDENTIFIED
User	Manage booking
Passenger	
Airline booking	
Airline enquery	
Permission	
Ticket	
Roles	

8.ER-Diagram



9.Functional dependencies and normalization

Normalization is the process of organizing data in a database to reduce redundancy and improve data integrity. It involves breaking down a database into smaller, more manageable tables and establishing relationships between them.

Normalization is typically divided into several levels of increasing complexity, each level building on the previous one. The most commonly used normalization levels are:

1.First Normal Form (1NF):

Eliminates repeating groups in a table by creating a separate table for each set of related attributes.
Ensures that each table has a primary key that uniquely identifies each row.

2.Second Normal Form (2NF):

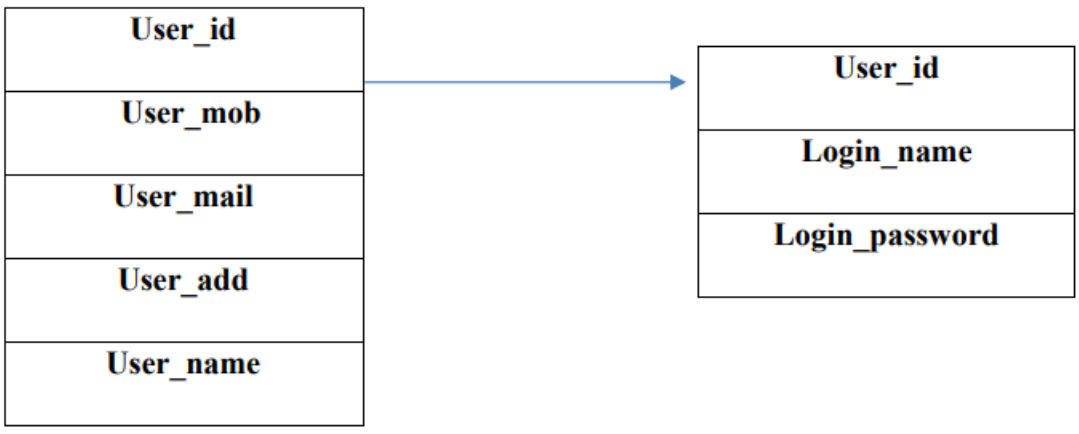
Ensures that all non-key attributes depend on the entire primary key, rather than just a part of it.
Eliminates redundancy by breaking down tables into smaller, more focused tables.

3.Third Normal Form (3NF):

Ensures that all non-key attributes are not dependent on other non-key attributes.
Further reduces redundancy by breaking down tables into smaller, more focused tables.
Normalization can be applied to an airline reservation system to ensure that data is organized efficiently and accurately. For example:

The flights table can be broken down into smaller tables, such as a flights schedule table, a seat inventory table, and a fares table.
Customer information can be stored in a separate table from reservation information to avoid redundancy and improve data integrity.
Loyalty program information can be stored in a separate table from customer information to avoid redundancy and ensure that changes to loyalty program rules do not affect customer data.
Overall, normalization helps to ensure that data is consistent and accurate, which is critical for an airline reservation system that deals with complex transactions and a high volume of data.

Schema Diagram for user and login relation



10.Data dictionary

Login relation :

Attribute	Data type	constraint	Remarks
User_id	number	Primary key	User's id
Login_name	Varchar(50)		Name of the user
Login_password	Varchar(50)		User's password

User relation:

Attribute	Data type	constraint	Remarks
User_id	number	Foreign key	User's id
User_mob	Varchar(50)		User's mobile number
User mail	Varchar(50)		User's mail
User_add	Varchar(50)		User's address
User_name	Varchar(50)		User's name

11.Database creation

```
create table user10(user_id number primary key,user_name varchar(50),user_Address
varchar(50),user_mob varchar(50),user_mail varchar(50));
insert into user10 values(0001,'Bhalaji','Ganapathi
Nagar',9171004532,'csebhalaji@gmail.com');
insert into user10
values(0002,'Ravikumar','Anaikattu',9786543202,'cseravikumar@gmail.com');
insert into user10 values(0003,'Nihil','Nagapattinam',9894100898,'csenihil@gmail.com');
insert into user10
values(0003,'Ramkishan','Villapuram',9842100878,'cseramkishan@gmail.com');
select *from user10;
desc user10;
create table login10( login_password varchar(50),login_name varchar(50),foreign
key(user_id) references user10(user_id));
insert into login10 values('bhalaji10','Bhalaji');
insert into login10 values('ravi10','Ravikumar');
```

insert into login10 values('nihil19','Nihil');
insert into login10 values('ramkishan7','Ramkishan');
select *from login10;

```
create table user10(user_id number primary key,user_name varchar(50),user_Address varchar(50),user_mob varchar(50),user_mail varchar(50));
insert into user10 values(0001,'Bhalaji','Ganapathi Nagar',9171004532,'csebhalaji@gmail.com');
insert into user10 values(0002,'Ravikumar','Anaikattu',9786543202,'cseravikumar@gmail.com');
insert into user10 values(0003,'Nihil','Nagapattinam',9894100898,'csenihil@gmail.com');
insert into user10 values(0003,'Ramkishan','Villapuram',9842100878,'cseramkishan@gmail.com');
select *from user10;
desc user10;
create table login10( login_password varchar(50),login_name varchar(50),foreign key(user_id) references user10(user_id));
insert into login10 values('bhalaji10','Bhalaji');
insert into login10 values('ravi10','Ravikumar');
insert into login10 values('nihil19','Nihil');
insert into login10 values('ramkishan7','Ramkishan');
select *from login10;
```

Results | Explain | Describe | Saved SQL | History

USER_ID	USER_NAME	USER_ADDRESS	USER_MOB	USER_MAIL
3	Nihil	Nagapattinam	9894100898	csenihil@gmail.com
1	Bhalaji	Ganapathi Nagar	9171004532	csebhalaji@gmail.com
4	Ramkishan	Villapuram	9842100878	cseramkishan@gmail.com
2	Ravikumar	Anaikattu	9786543202	cseravikumar@gmail.com

4 rows returned in 0.22 seconds Download

```
create table login10( login_password varchar(50),login_name varchar(50),foreign key(user_id) references user10(user_id));
insert into login10 values(0001,'bhalaji10','Bhalaji');
insert into login10 values(0002,'ravi10','Ravikumar');
insert into login10 values(0003,'nihil19','Nihil');
insert into login10 values(0004,'ramkishan7','Ramkishan');
select *from login10;
```

Results | Explain | Describe | Saved SQL | History

USER_ID	LOGIN_PASSWORD	LOGIN_NAME
1	bhalaji10	Bhalaji
3	nihil19	Nihil
1	bhalaji10	Bhalaji
2	ravi10	Ravikumar
4	ramkishan7	Ramkishan

5 rows returned in 0.01 seconds Download

12.Database connectivity

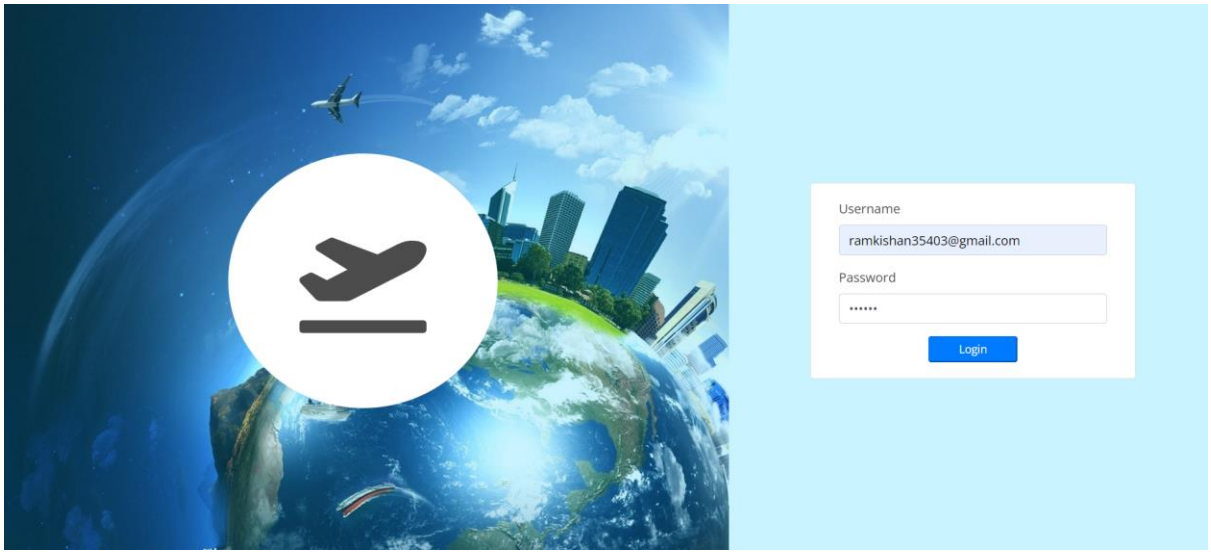
Sample code:

```
<!DOCTYPE html>
<html lang="en">
   <?php
   session_start();
   ob_start();
   include('header.php');
   include('admin/db_connect.php');

            $query = $conn->query("SELECT * FROM system_settings limit 1")->fetch_array();
            foreach ($query as $key => $value) {
                     if(!is_numeric($key))
                              $_SESSION['setting_'.$key] = $value;
            }
   ob_end_flush();
   ?>

   <style>
            header.masthead {
                      background: url(assets/img/<?php echo $_SESSION['setting_cover_img'] ?>);
                      background-repeat: no-repeat;
                      background-size: cover;
                     }
   </style>
   <body id="page-top">
      <!-- Navigation-->
```

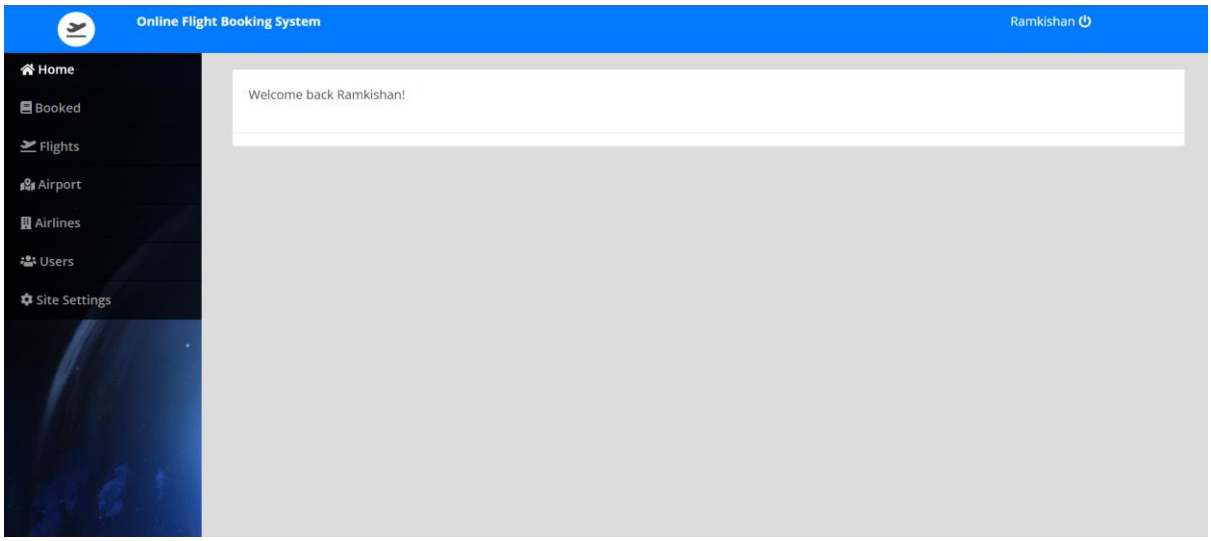
```
      <div class="toast" id="alert_toast" role="alert" aria-live="assertive" aria-atomic="true">
      <div class="toast-body text-white">
      </div>
    </div>
      <nav class="navbar navbar-expand-lg navbar-light fixed-top py-3" id="mainNav">
        <div class="container">
          <a class="navbar-brand js-scroll-trigger" href="./"><?php echo $_SESSION['setting_name'] ?></a>
          <button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation"><span class="navbar-toggler-icon"></span></button>
          <div class="collapse navbar-collapse" id="navbarResponsive">
            <ul class="navbar-nav ml-auto my-2 my-lg-0">
              <li class="nav-item"><a class="nav-link js-scroll-trigger" href="index.php?page=home">Home</a></li>
              <li class="nav-item"><a class="nav-link js-scroll-trigger" href="index.php?page=flights"></span>Flight
List</a></li>
              <li class="nav-item"><a class="nav-link js-scroll-trigger" href="index.php?page=about">About</a></li>


            </ul>
          </div>
        </div>
      </nav>

      <?php
      $page = isset($_GET['page']) ?$_GET['page'] : "home";
      include $page.'.php';
      ?>


<div class="modal fade" id="confirm_modal" role='dialog'>
   <div class="modal-dialog modal-md" role="document">
    <div class="modal-content">
     <div class="modal-header">
     <h5 class="modal-title">Confirmation</h5>
    </div>
    <div class="modal-body">
     <div id="delete_content"></div>
    </div>
    <div class="modal-footer">
     <button type="button" class="btn btn-primary" id='confirm' onclick="">Continue</button>
     <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
    </div>
    </div>
   </div>
  </div>
  <div class="modal fade" id="uni_modal" role='dialog'>
   <div class="modal-dialog modal-md" role="document">
    <div class="modal-content">
     <div class="modal-header">
     <h5 class="modal-title"></h5>
    </div>
    <div class="modal-body">
    </div>
    <div class="modal-footer">
     <button type="button" class="btn btn-primary" id='submit' onclick="$('#uni_modal form').submit()">Save</button>
     <button type="button" class="btn btn-secondary" data-dismiss="modal">Cancel</button>
    </div>
    </div>
   </div>
  </div>
  <div class="modal fade" id="uni_modal_right" role='dialog'>
   <div class="modal-dialog modal-full-height  modal-md" role="document">
    <div class="modal-content">
     <div class="modal-header">
     <h5 class="modal-title"></h5>
     <button type="button" class="close" data-dismiss="modal" aria-label="Close">
      <span class="fa fa-arrow-righ t"></span>
```

```
        </button>
      </div>
      <div class="modal-body">
      </div>
      </div>
    </div>
  </div>
  <div id="preloader"></div>
      <footer class="bg-light py-5">
        <div class="container">
          <div class="row justify-content-center">
            <div class="col-lg-8 text-center">
              <h2 class="mt-0">Contact us</h2>
              <hr class="divider my-4" />
            </div>
          </div>
          <div class="row">
            <div class="col-lg-4 ml-auto text-center mb-5 mb-lg-0">
              <i class="fas fa-phone fa-3x mb-3 text-muted"></i>
              <div><?php echo $_SESSION['setting_contact'] ?></div>
            </div>
            <div class="col-lg-4 mr-auto text-center">
              <i class="fas fa-envelope fa-3x mb-3 text-muted"></i>
              <!-- Make sure to change the email address in BOTH the anchor text and the link target below!-->
              <a class="d-block" href="mailto:<?php echo $_SESSION['setting_email'] ?>"><?php echo
$_SESSION['setting_email'] ?></a>
            </div>
          </div>
        </div>
        <br>
        <div class="container"><div class="small text-center text-muted"> <?php echo $_SESSION['setting_name'] ?> | <a
href="https://www.campcodes.com" target="_blank">CampCodes</a></div></div>
      </footer>

    <?php include('footer.php') ?>
  </body>

  <?php $conn->close() ?>

</html>
```

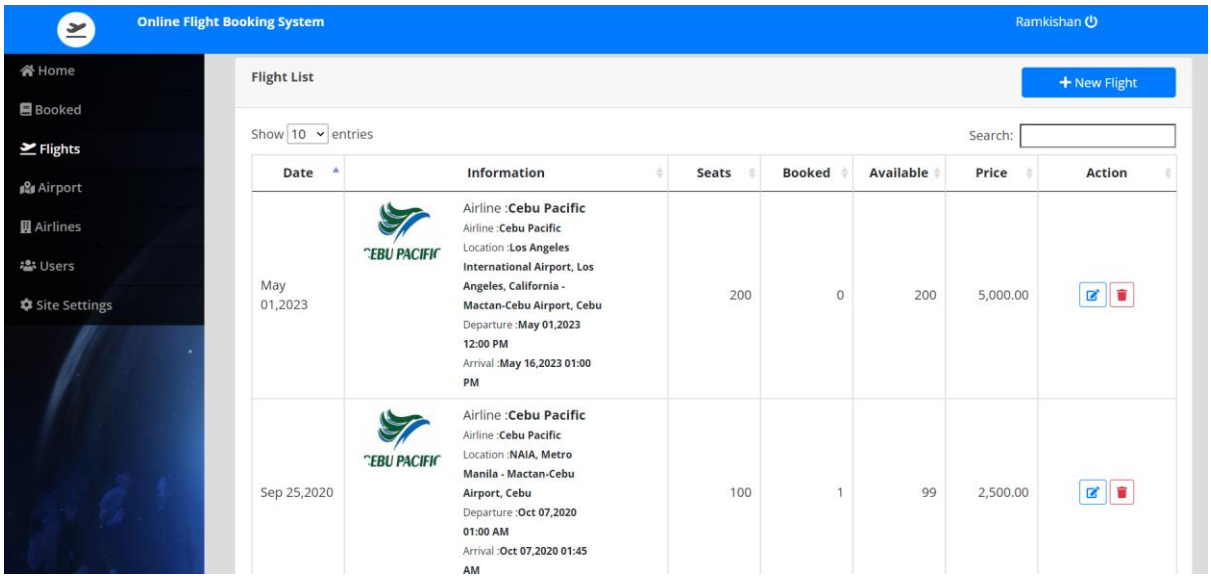
Login page:



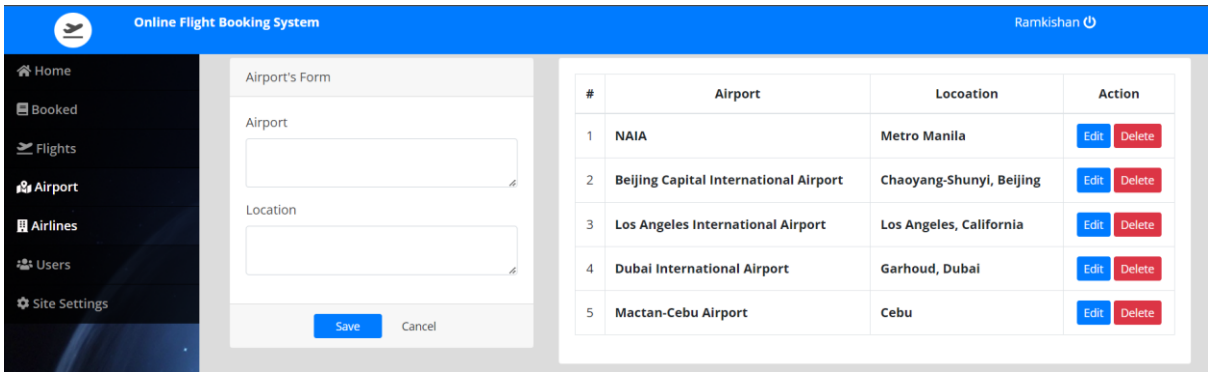
Home page:



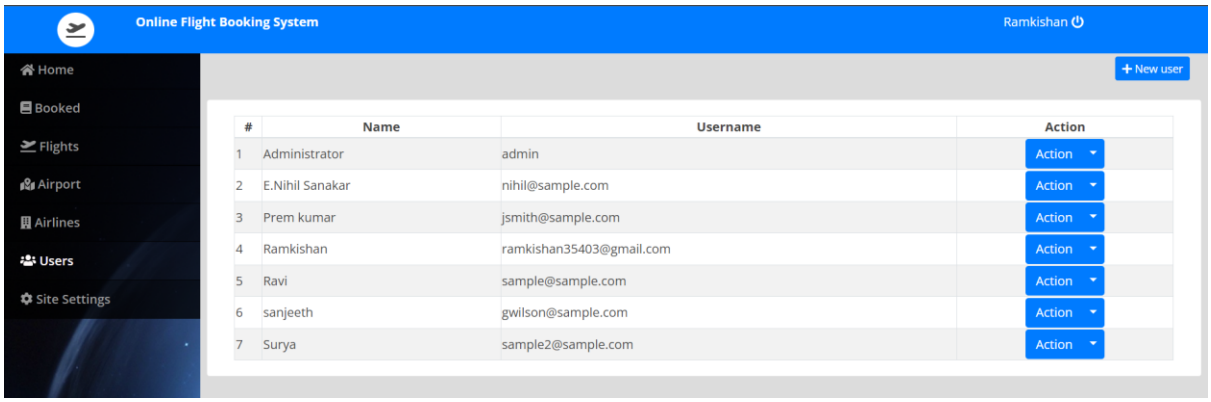
Flights availability:



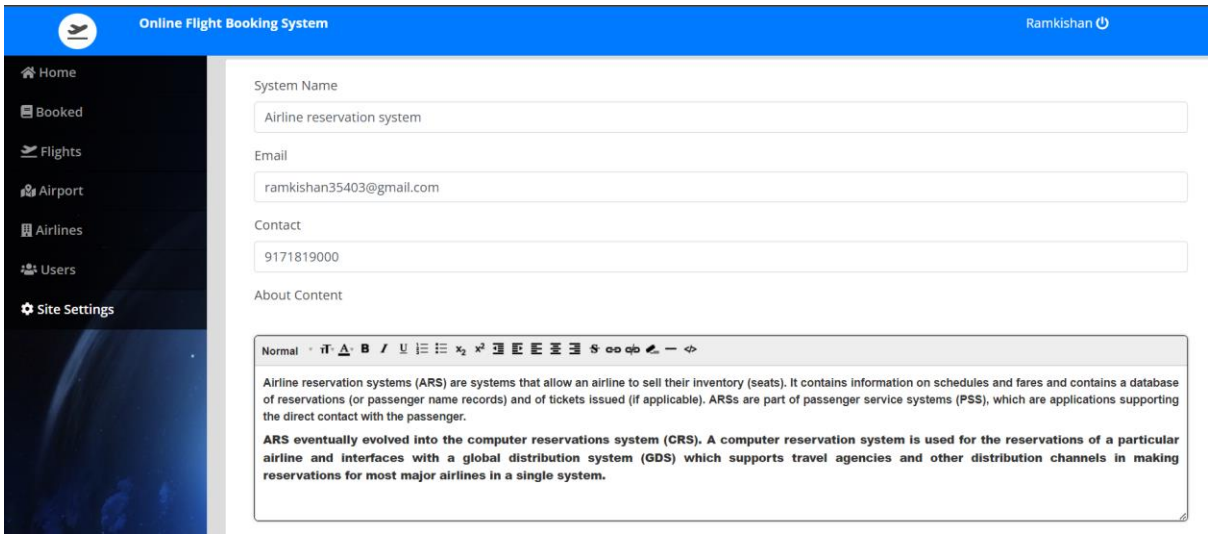
Airports:



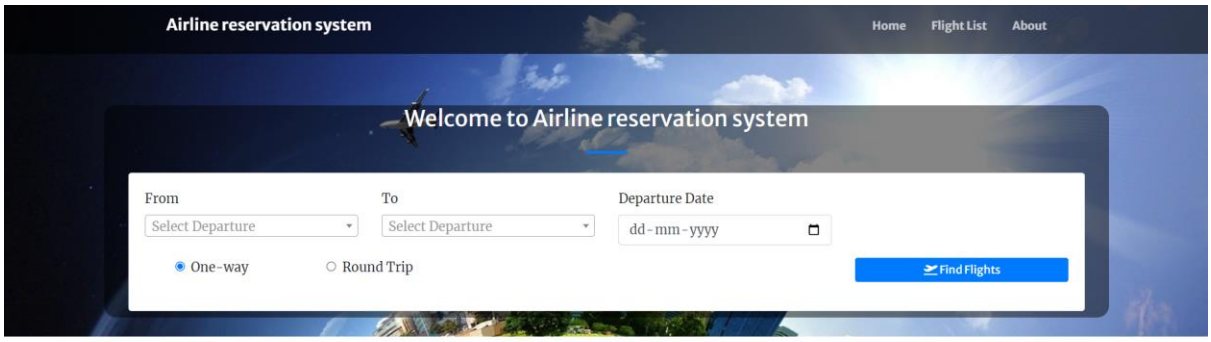
Users:

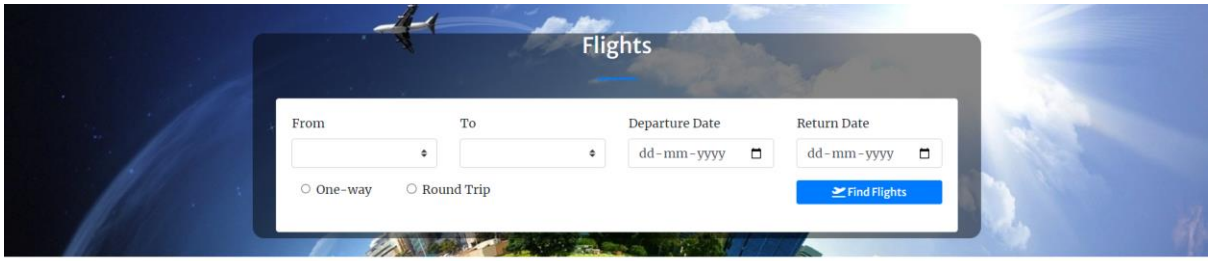


Site settings:



Filght bookings:





About:

Airline reservation systems (ARS) are systems that allow an airline to sell their inventory (seats). It contains information on schedules and fares and contains a database of reservations (or passenger name records) and of tickets issued (if applicable). ARSs are part of 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.

Contact us

9171819000

ramkishan35403@gmail.com

Airline reservation system | CampCodes

13.Conclusion

In conclusion, an airline reservation system is a critical software application that enables customers to search, book, and manage flights and travel itineraries. The system must handle a large volume of data and complex transactions in real-time, ensuring that flights are scheduled, seats are assigned, and payments are processed efficiently and accurately.

To achieve this, the system requires robust functionality, including flight search and booking, reservation management, check-in and boarding, loyalty program integration, payment and security, customer support, and ancillary services. These functions must work seamlessly together to provide customers with a smooth and reliable experience.

Normalization is a critical part of the development process for an airline reservation system, as it helps to ensure that data is organized efficiently and accurately, reducing redundancy and improving data integrity. By breaking down tables into smaller, more focused tables, normalization makes it easier to manage and maintain the system, reducing the risk of errors and improving system performance.

Overall, a well-designed airline reservation system is essential for the success of an airline, providing a competitive advantage by improving the customer experience, reducing costs, and increasing revenue.

14.Reference

- https://www.w3schools.com/html/html_layout.asp
- https://api.jquery.com/
- https://www.javatpoint.com/run-php-code-in-xampp