SDJ INTERNATIONAL COLLEGE

# Bachelor of Computer Application (BCA) Programme

## Minor Project Report

BCA Sem V
AY 2022-23

## *Online Flight Booking System*

*by*

| Exam No. | Roll No. | Name of Student |
|---|---|---|
| 4113 | 091 | Kakadiya Shivangi A. |

**Project Guide by :**
Prof. Bhumika Patel

# *I N D E X*

# Introduction

## 1.1 Project Summary

Online Flight Booking System provides details about flight schedules and its fare tariffs, Passenger reservation and ticket records. OFBS operates up to 18 domestic flights of America. It saves time as it allows online procedure as users no longer have to wait in queues to book their tickets.

Admin is the main authority who can do flight addition, see feedbacks, solve flight issues, receive payment and do flight deletion. In general, this website would be designed to perform like any other flight booking websites available online.

## 1.2 Project Profile

| | |
|---|---|
| **Project Title:** | Online Flight Booking System |
| **Definition :** | Online Flight Booking System will provides a platform for users to book flights online and ease their future travels. |
| **Developed For :** | SDJ International College, Vesu, Surat |
| **Project Guide(s):** | Prof. Bhumika Patel |
| **Front End:** | PHP |
| **Scripting language :** | PHP, CSS, BOOTSTRAP, JAVASCRIPT |
| **Back End :** | Xampp Server |
| **Operating System:** | Microsoft Windows 7 or higher |
| **Tools used for ERD & DFD** | Visual Studio, Notepad ++, Xampp , Chrome |
| **Submitted By** | Kakadiya Shivangi Anilbhai (4113) |

# Environment Description

## 2.1 Hardware and Software Requirements

Online flight booking system requires following technical specifications to run properly and efficiently.

SERVER SIDE:

- **Hardware Requirements**
  - Intel(R) Core (TM) i3-4005U CPU @ 1.70GHz 1.70GHz
  - 2.00GB RAM

- **Soft ware Requirements**
  Windows 7 Ultimate or higher
  MySQL
  XAMPP
  Bootstrap, CSS, Javascript.

CLIENT SIDE:

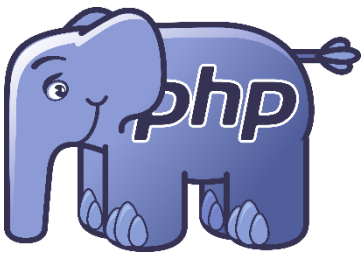
- **Hardware Requirements**

  Intel(R) Core (TM) i3-4005U CPU @ 1.70GHz 1.70GHZ
  2.00GB RAM

- **Soft ware Requirements**
  Windows 7 Ultimate or higher
  Browsers : Mozila firefox, Google Chrome.

## 2.2 Technologies Used

Front End: PHP
PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.
What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server, generating HTML which is then sent to the client. The client would receive the results of running that script, but would not know what the underlying code was. You can even configure your web server to process all your
HTML files with PHP, and then there's really no way that users can tell what you have up your sleeve.
The best things in using PHP are that it is extremely simple for a newcomer, but offers many advanced features for a professional programmer.



## Back End: MySQL

MySQL runs on virtually all platforms, including Linux, UNIX, and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web-based applications and online publishing and is an important component of an open source enterprise stack called LAMP. LAMP is a Web development platform that uses Linux as the operating system, Apache as the Web server, MySQL as the relational database management system and PHP as the object-oriented scripting language.MySQL is an essential part of almost every open source PHP application. Good example for PHP/MySQL-based script are PHPBB.

## XAMPP:

XAMPP stands for Cross-Platform (X), Apache (A), MySQL (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing purposes. Everything you need to set up a web server - server application (Apache), database (MySQL), and scripting language (PHP) - is included in a simple extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use same components as XAMPP, it makes transitioning from a local test server to a live server is extremely easy as well. XAMPP has four primary components:

• Apache: Apache is the actual web server application that processes and delivers web content to a computer. Apache is the most popular web server online, powering nearly 54% of all websites.

• MySQL: Every web application, howsoever simple or complicated, requires a database for storing collected data. MySQL, which is open source, is the world's most popular database management system. It powers everything from hobbyist websites to professional platforms like WordPress. You can learn how to master PHP with this free MySQL database for beginner's course.

• PHP: PHP stands for Hypertext Preprocessor. It is a server-side scripting language that powers some of the most popular websites in the world, including WordPress and Facebook. It is open source, relatively easy to learn, and works perfectly with MySQL, making it a popular choice for web developers.

## Bootstrap:

Bootstrap is a free and open-source front-end web framework for designing websites and web applications. It contains HTML- and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions.
It aims to ease the development of dynamic website and web application.

Bootstrap is a front end web framework, that is, an interface for the user, unlike the server-side code which resides on the "back end" or server.
Bootstrap provides a set of stylesheets that provide basic style definitions for all key HTML components. These provide a uniform, modern appearance for formatting text, tables and form elements.

• CSS:


Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications. Before CSS, nearly all presentational attributes of HTML documents were contained within the HTML markup. All font colors, background styles, element alignments, borders and sizes had to be explicitly described, often repeatedly, within the HTML. CSS lets authors move much of that information to another file, the style sheet, resulting in considerably simpler HTML.

JavaScript:


JavaScript is a high-level, dynamic, untyped, and interpreted programming language. It has been standardized in the ECMAScript language specification. Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Web content production; the majority of websites employ it, and all modern Web browsers support it without the need for plug-ins. JavaScript is prototype-based with first-class functions, making it a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. It has an API for working with text, arrays, dates and regular expressions, but does not include any I/0, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

Ajax:

Ajax is a client-side script that communicates to and from a server/database without the need for a postback or a complete page refresh. (The method of exchanging data with a server, and updating parts of a web page - without reloading the entire page).

# System Analysis and Planning

## 3.1 Existing System and its Drawbacks

The existing system is that the passenger must fill up the data manually and must submit it to the reservation counter. It may take a lot of time to process it and to book the flight. Therefore, there is wastage of time. Since the data is entered manually, the probability of error or mistakes is high.

- Cannot Upload and Download the latest updates.
- No use of Web Services and Remoting
- Risk of mismanagement and of data when the project is under development.
- Less Security.
- Its difficult to update, delete, or view the data due its manual nature.
- The existing system consumes a lot of time causing inconveniencing to customers and the staff.
- No proper coordination between different Applications and Users.
- Increasing number of passengers leads to difficulty in maintaining and retrieving detail.
- Fewer Users - Friendly

## 3.2 Feasibility Study

Not everything imaginable is feasible, not even in software, evanescent at it appears to outsiders. The feasibility study is done to decide whether we should undergo in taking project or not. And select the project only if it is feasible in terms of cost, time, technology and resources and also Operational feasibility is involved in considering a project.
On the contrary, software feasibility has four solid dimensions.

1. Technological feasibility:
Technological feasibility includes various aspects such as:

Whether the project is technically feasible?
Our project is technically feasible. The technology we are using or implementing in our software is easily available and is user friendly. It is also compatible with the current computer system used nowadays.

Can it be reduced to a level matching the applications need?
Yes, the software can be reduced to the level matching application needs. The computer language we are using is complete advanced one though; it has facility of reducing to the level of our application.

2. Financial Feasibility:
It includes basic two aspects:

Is it financially feasible?
The software which we are developing is financially feasible, as it requires a minimum of desktop computer with basic peripherals and two easily available software.

Can development be completed at a cost of the software organization, it's client, or the market can afford?
Yes,the software we are developing can be completed at the cost of organization's clients. Market can easily afford the software as its costing is not going to be unfeasible.

3. Time Feasibility:
It includes the main aspect of being the markets competition within the time.
The software is fully compatible with the soft-wares present in market related to club management system. There are many other extra feature which beat the market that to within the given time limit.

4. Resources Feasibility:
1. The resources available in company are sufficient to develop the software. They are fully updated and are ready to use. Thus, the software is feasible from resources point of view.

### 3.3 Requirement Gathering and Analysis

The Software Requirements Specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints. appropriate validation criteria, and other data pertinent to requirements.

Hardware Requirements -

For the hardware requirements like memory restrictions, cache size, the processor, RAM size etc... those are required for the software to run.

MINIMUM Hardware Requirements
Processor Pentium IV Hard Drive 100 GBRAM 1 Gb

PREFERED HARDWARE REQUIREMENTS
Processor Core i3Hard Disk Drive 500 GB RAM 4 GB

Software Requirements –

Any window based operating system with DOS support are primary requirements for software development. Windows 7 and up are required. The system must be connected vie LAN and connection to internet is mandatory.

Other Requirements:

- Security
- Portability
- Correctness
- Efficiency
- Flexibility
- Reusability

Performance requirements:

- User Satisfaction: The system is such that it stands up to the user expectations.
- Response Time: The response of all operations is good.
- Error Handling: Response to user errors and undesired situation has been taken care of to ensure that the system operates without halting.
- Safety and Robustness: The system is able to avoid or tackle disastrous action. In other words it should be foul proof.
- Portable: The software should not be architecture specific. It should be easily transferable to other platforms if needed.
- User Friendliness: The system is easy to learn and understand. A native user can also use the system effectively, without any difficulties

# Processed System

## 4.1 Scope

The airline booking website is an application stored in the user server. The purpose of the website is to resolve the client to allow website users to perform tasks related to booking an airline flight.

- The system enables to perform the following functions:
- Automation of flight operations
- Automation of ticketing / seat booking
- confirmation system
- Cancellation
- Improved and optimized service

## 4.2 Project modules

- Flight Management Module: Used for managing the Flight details.
- Payment Module: Used for managing the details of Payment
- Feedback Module : Used for managing the details of Feedbacks.
- Ticket Management Module: Used for managing the information and details of Ticket.
- Booking Module: Used for managing the Booking details.
- Passenger Module: Used for managing the Passenger information.
- Login Module: Used for managing the login details.
- Users Module: Used for managing the users of the system.

### 4.3 Module vise objectives/functionalities Constraints

There are a number of factors in the client's environment that may restrict the choices of a designer. Such factors include standards that must be followed, resource limits, operating environment, reliability and security requirements and policies that may have an impact on the design of the system.

Standard Compliances:
This specifies the requirement for standards the system must follow. The standards may include the report format and accounting properties

Hardware Limitations:
Hardware limitations can include the types of machine to be used, operating system available on the system, languages support and limits on primary and secondary storage.

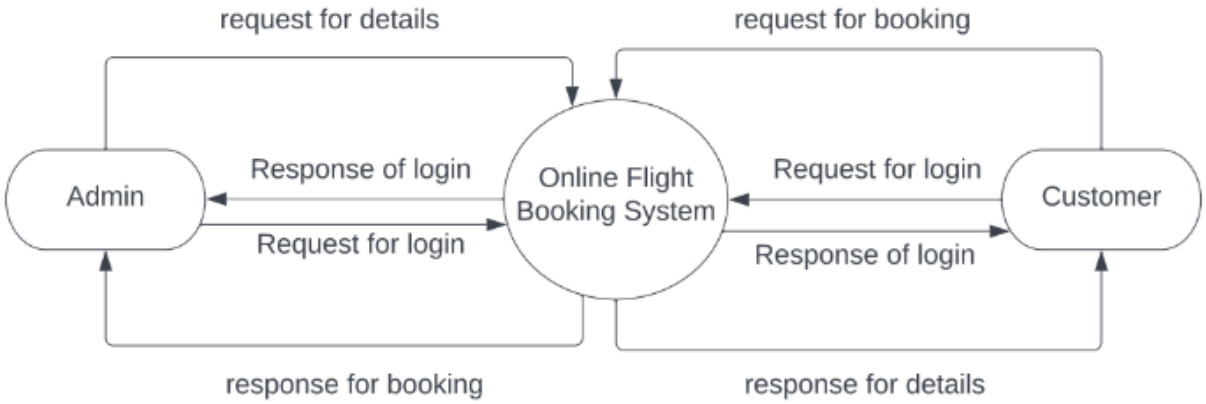
Reliability and Fault Tolerance:
Fault tolerance requirement can be place a constraint on how the system is to be designed. Recovery requirements are often on integral part here, detailing what the system should do if some failure occurs to ensure certain properties. Reliability requirements are very important for critical application.

Security:
Security requirements are particularly significant in defense system and database system. They place restrictions on the use of certain commands, control access to data, provide different kinds of access requirements for different people, require the use of passwords and cryptography techniques and maintain a log of activities in the system.

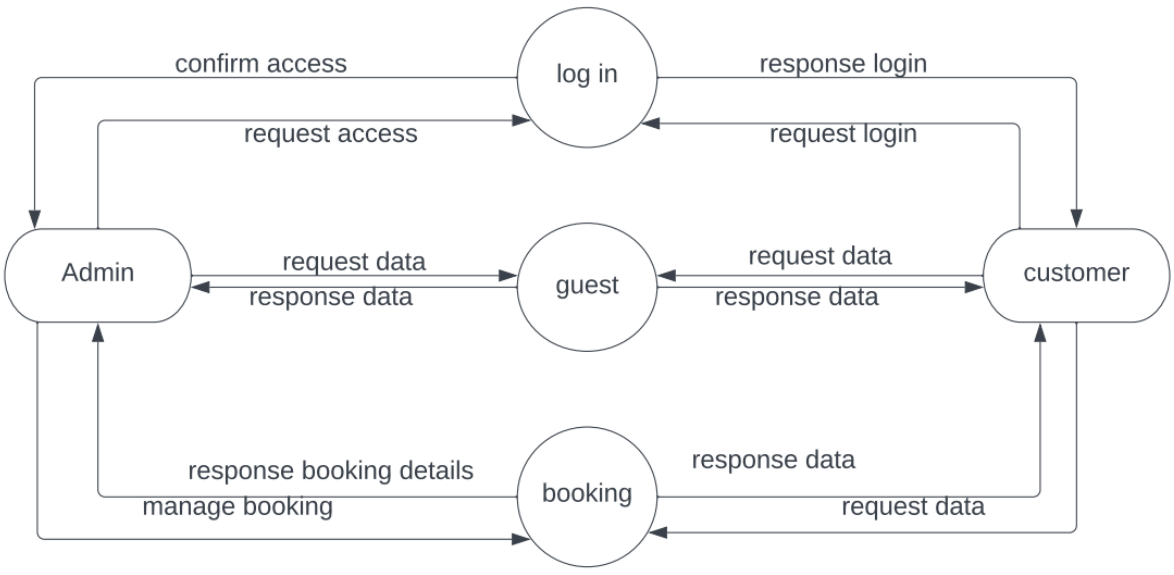
# Detail Planning

## 5.1 Data Flow Diagram / UML

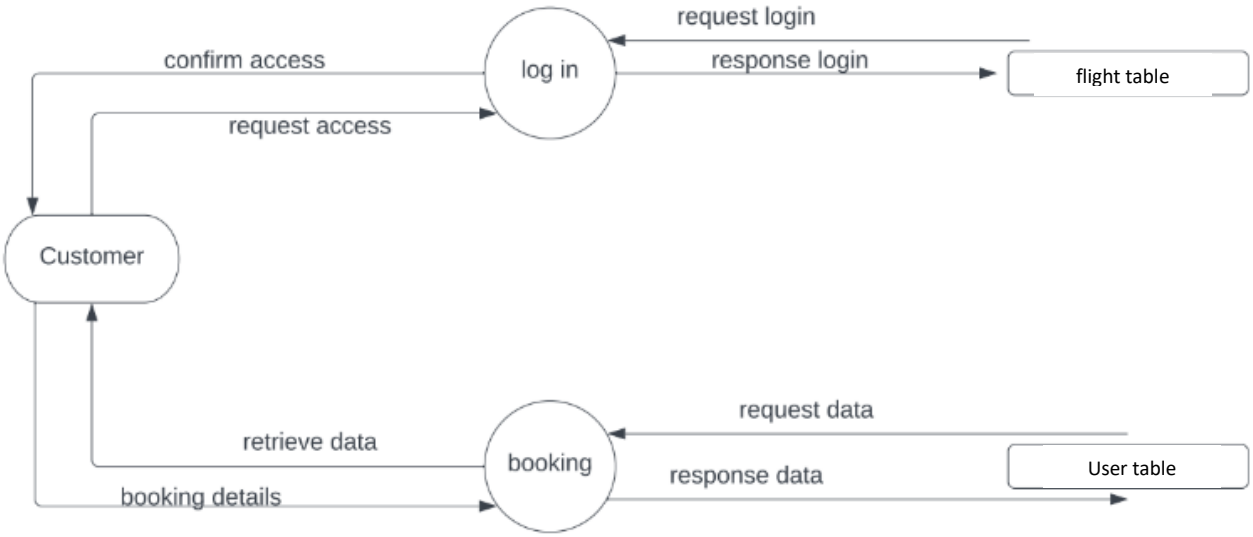
### Context-level:



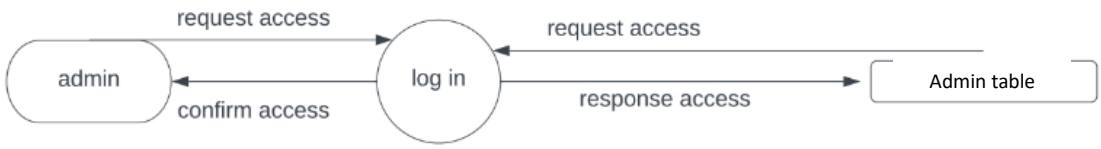

### 1st Level (admin) Diagram:



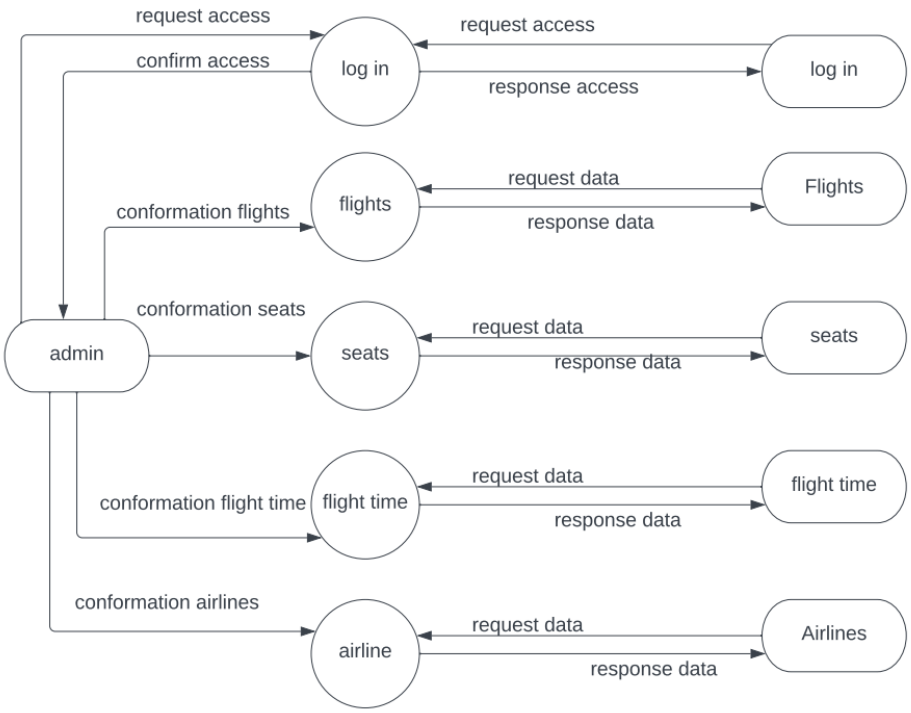
1st Level (user) Diagram:




**2nd level login(admin) Diagram:**

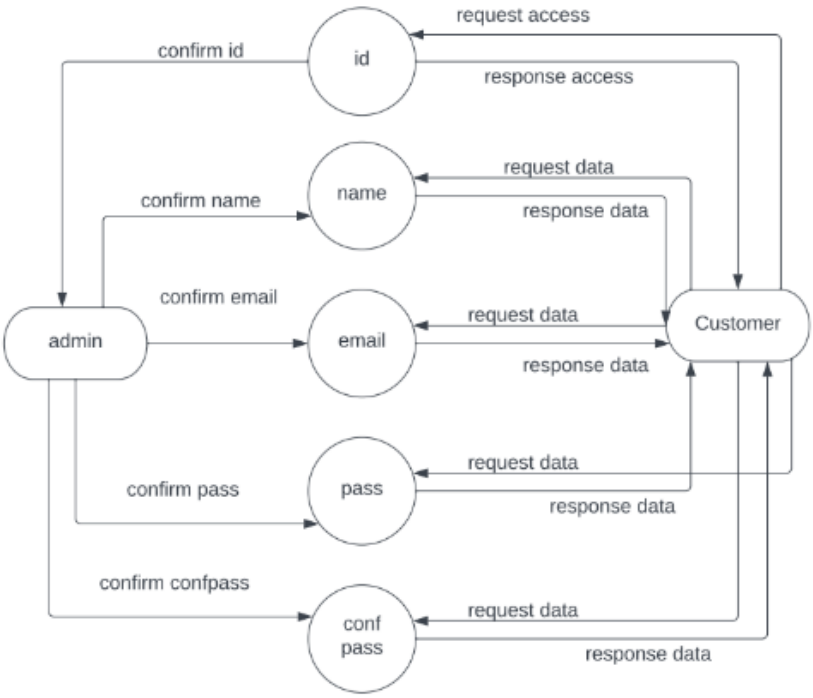


## 2nd level booking(admin) Diagram:

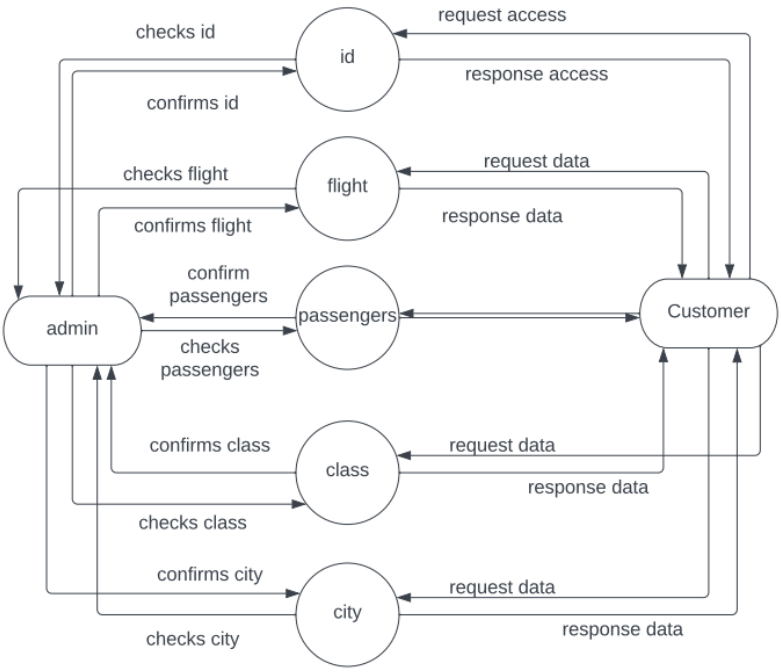



## 2nd level login(user) Diagram:

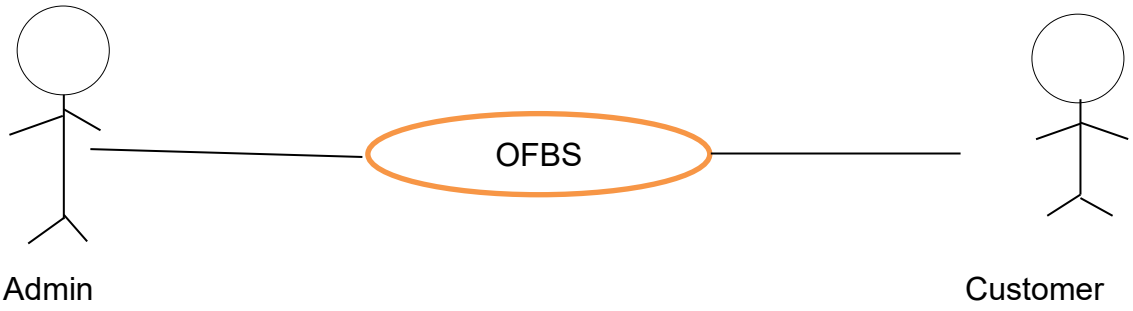

## 2nd level login(user) Diagram:



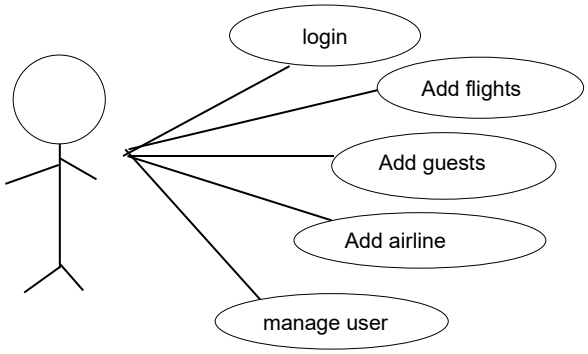

## 2nd level booking(user) Diagram:



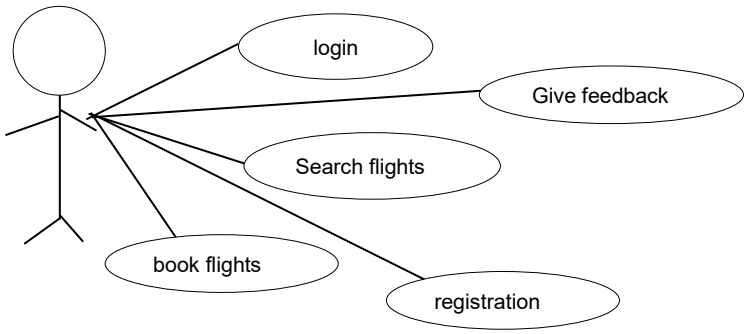
**UML diagram:**



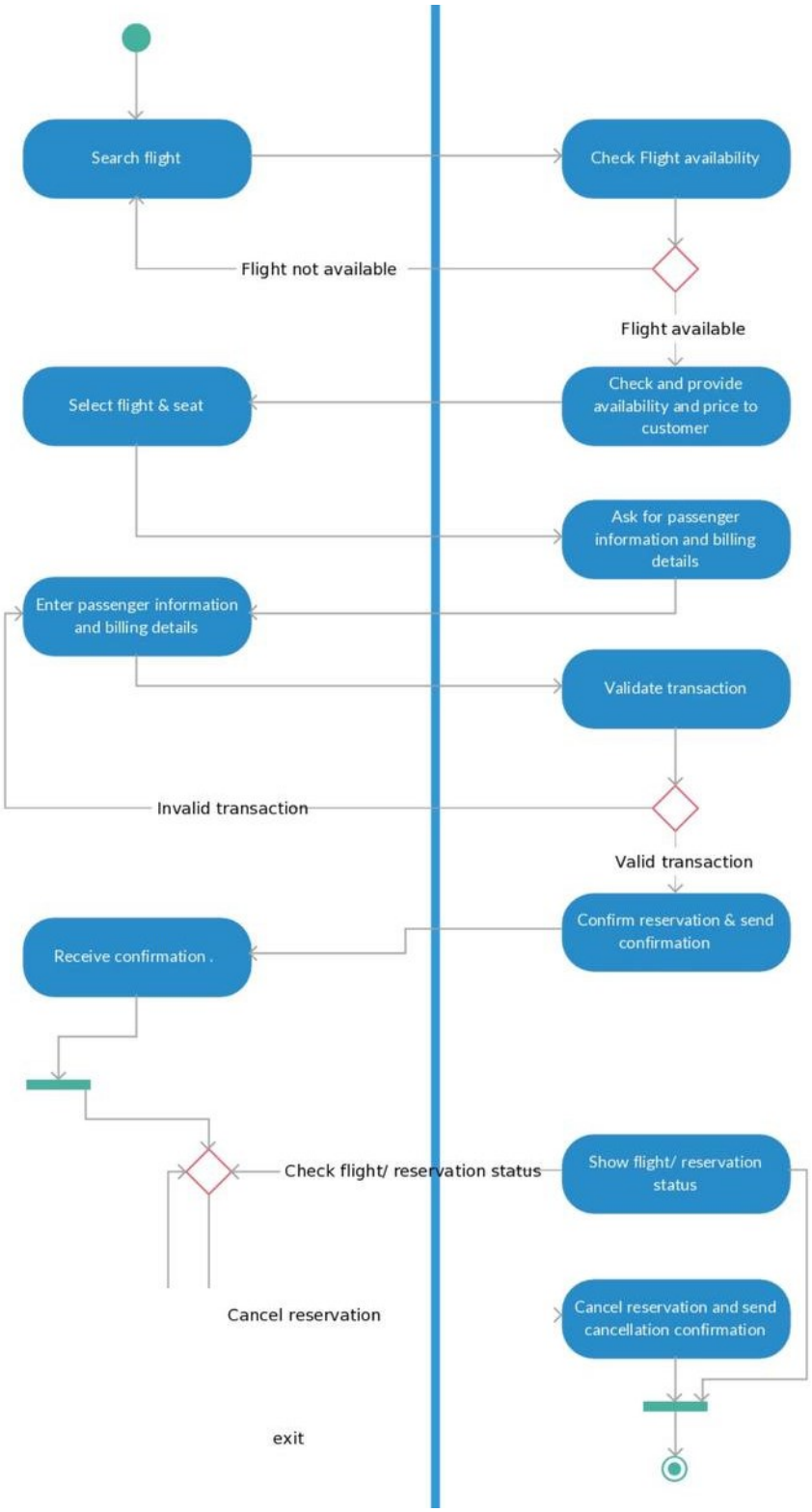

**Admin Side:**




**User side:**



## 5.2 Process Specification / Activity Flow Diagram



### 5.3 Data Directories

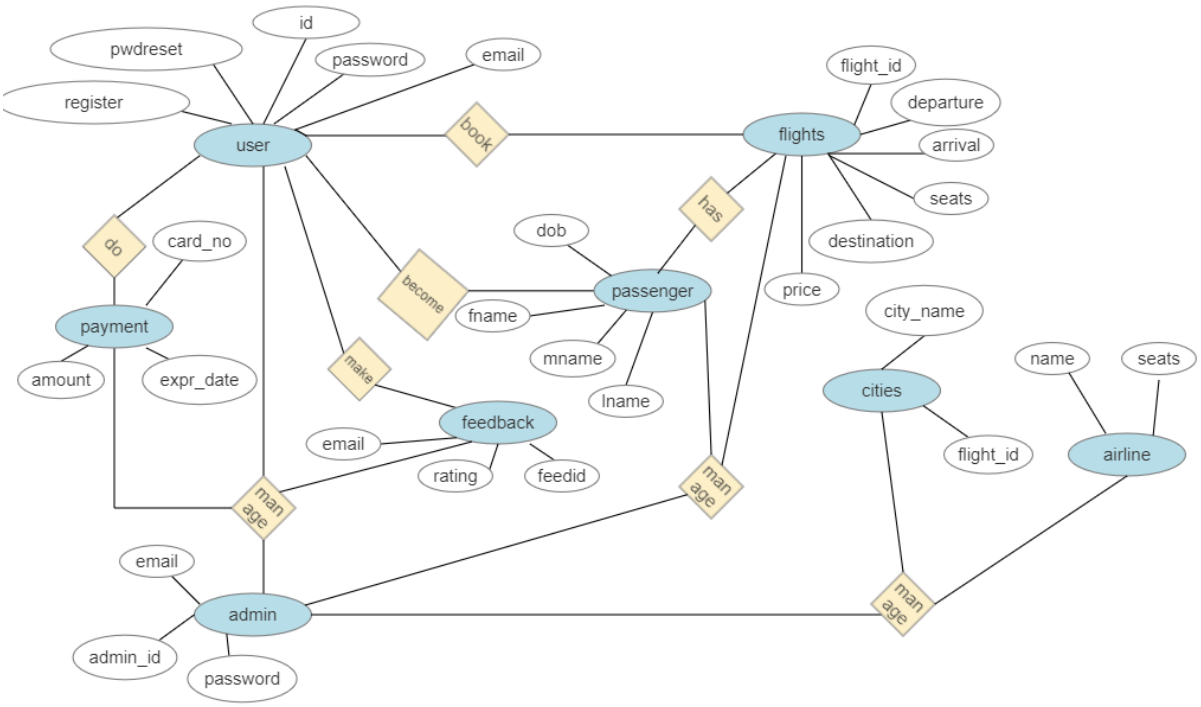
| USER | |
|---|---|
| Alias | Null |
| Where used/How used | To retrieve or store user details |
| Description | name + email + userid + password + confirm password |
| Supplementary information | Userid must be unique |

| ADMIN | |
|---|---|
| Alias | Null |
| Where used/How used | To retrieve or store admin details |
| Description | userid + password |

| Flights | |
|---|---|
| Alias | Null |
| Where used/How used | To retrieve or store details |
| Description | flightid + arrival + departure +destination + airline + seats +duration |
| Supplementary information | Flightid must be unique |

| List flights | |
|---|---|
| Alias | Null |
| Where used/How used | To retrieve or store details |
| Description | flightid + arrival + departure +destination + airline + seats +duration |

## 5.4 Entity-Relationship Diagram / Class Diagram



# System Design

## 6.1 Database Design

**Table: admin**
**Description: This table gives detail about admin information**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| admin_id | int(11) | Primary key | admin id |
| admin_uname | varchar(20) | not null | admin username |
| admin_email | varchar(50) | not null | admin email |
| admin_pwd | varchar(100) | not null | admin password |

**Table: airline**
**Description: This table gives detail about airline information**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| airline_id | int(11) | Primary key | airline id |
| name | varchar(20) | not null | name of airline |
| seats | int(11) | not null | seats |

**Table: city**
**Description: This table gives detail about city information**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| city | varchar(50) | Not null | name of city |

**Table: flight**
**Description: This table gives detail about flight information**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| flight_id | int(11) | Primary key | flight id |
| admin_id | int(11) | Foreign key | admin id |
| arrival | datetime | Not null | arrival |
| departure | datetime | Not null | departure |
| destination | varchar(20) | Not null | destination |
| source | varchar(20) | Not null | source |
| airline | varchar(20) | Not null | airline |
| seats | varchar(110) | Not null | seats |
| duration | varchar(20) | Not null | duration |
| price | int(11) | Not null | price |
| status | varchar(6) | Null | flight status |
| issue | varchar(50) | Null | issues |
| last_seat | int(11) | Null | last seat |
| bus_seats | varchar(5) | Null | bus seats |

**Table: passenger_profile**
**Description: This table gives detail about passenger profile.**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| passenger_id | int(11) | Primary key | passenger id |
| user_id | int(11) | Foreign key | used id |
| flight_id | int(11) | Foreign key | flight id |
| mobile | varchar(10) | Not null | mobile |
| dob | datetime | Not null | date of birth |
| f_name | varchar(20) | Null | first name |
| m_name | varchar(20) | Null | middle name |
| l_name | varchar(20) | Null | last name |

**Table: payment**
**Description: This table gives detail about payment.**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| card_no | varchar(16) | Primary key | card number |
| user_id | int(11) | Foreign key | user id |
| flight_id | int(11) | Foreign key | flight id |
| expire_date | varchar(5) | Not null | expiry date |
| amount | int(11) | Not null | amount |

**Table: pwdreset**
**Description: This table gives detail about password reset.**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| pwd_reset_id | int(11) | Primarykey | Password reset id |
| pwd_reset_email | varchar(50) | Not null | Password reset email |
| pwd_reset_selector | varchar(80) | Not null | Password reset selector |
| pwd_reset_token | varchar(120) | Not null | Password reset token |
| pwd_reset_expires | varchar(20) | Not null | Password reset expires |

**Table: tickets**
**Description: This table gives detail about tickets.**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| ticket_id | int(11) | Primarykey | ticket id |
| passenger_id | int(11) | Foreign key | Passenger id |
| flight_id | int(11) | Foreign key | Flight id |
| user_id | int(11) | Foreign key | User id |
| seat_no | varchar(50) | Not null | Seat number |
| cost | varchar(50) | Not null | Cost |
| class | varchar(50) | Not null | class |

**Table: user**
**Description: This table gives detail about user information.**

| Field Name | Field Type | Constraint | Description |
|---|---|---|---|
| user_id | int(11) | Primarykey | uder id |
| username | int(20) | Not null | User name |
| email | int(50) | Not null | Email |
| password | int(100) | Not null | password |

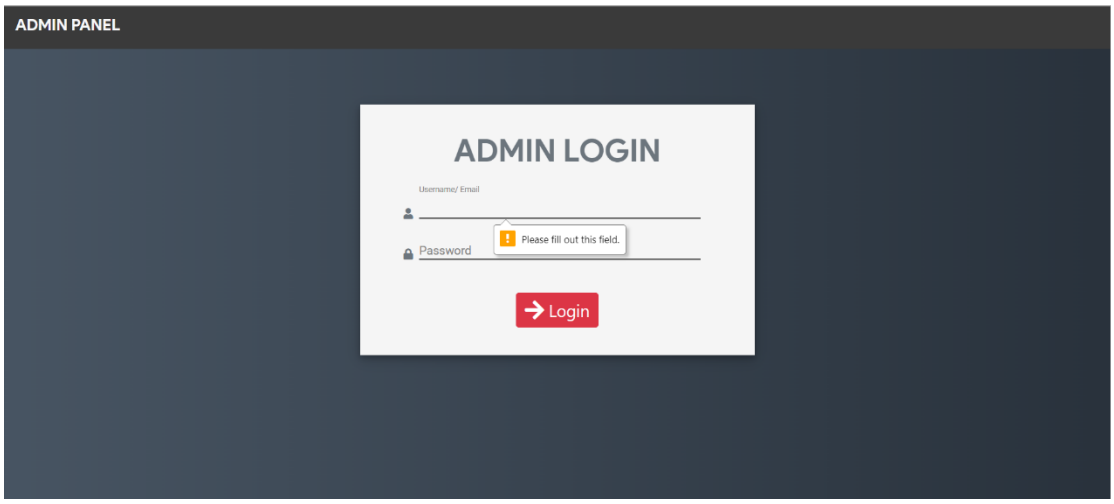
### 6.2 database structure

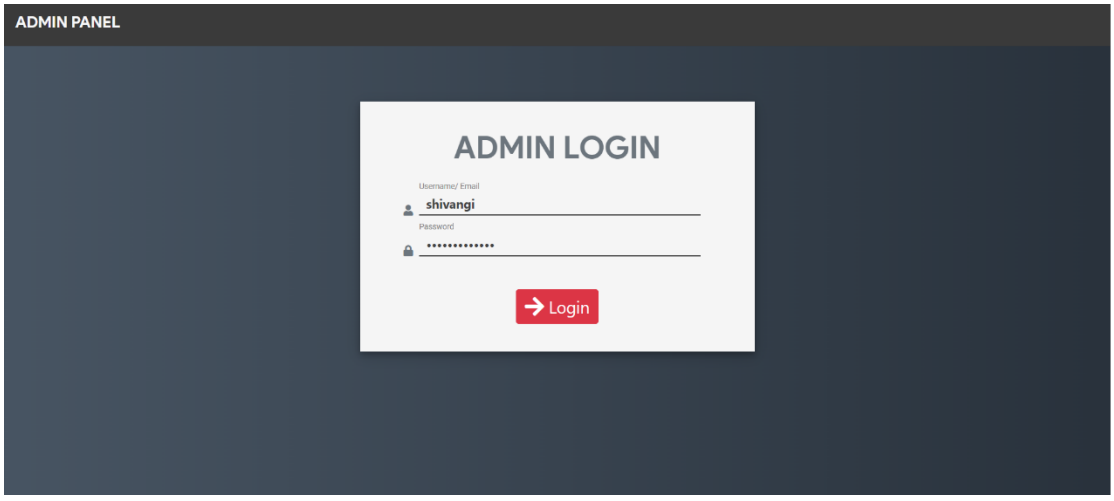
- Admin:
  - all_flights.php
  - amtcnt.php
  - flight.php
  - flightsent.php
  - footer.php
  - header.php
  - index.php
  - list_airlines.php
  - login.php
  - pass_list.php
  - psngrent.php

- User
  - about.php
  - book_flight.php
  - contact.php
  - create_new_pwd.php
  - e_ticket.php
  - feedback.php
  - index.php
  - login.php
  - my_flights.php
  - pass_form.php
  - pay_success.php
  - payment.php
  - register.php
  - reset_pwd.php
  - ticket.php

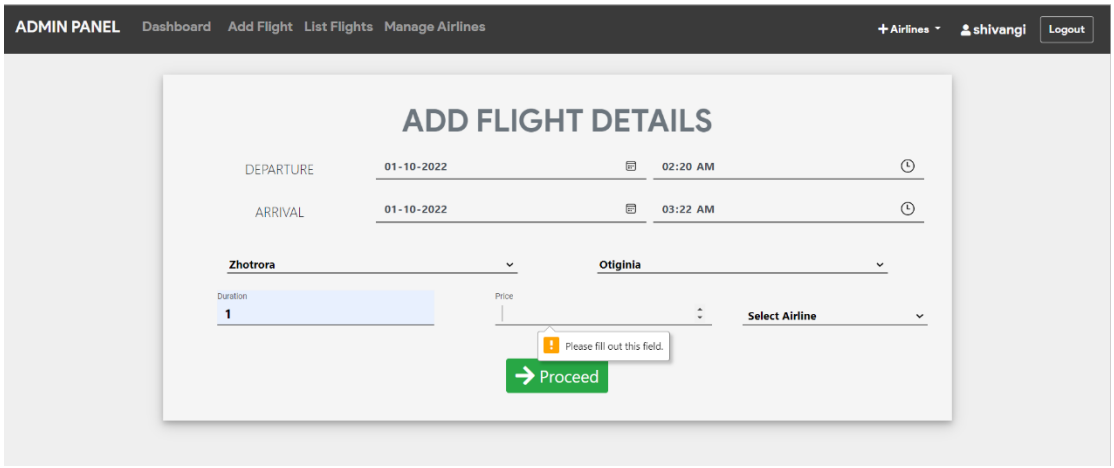
## 6.3 Input Design

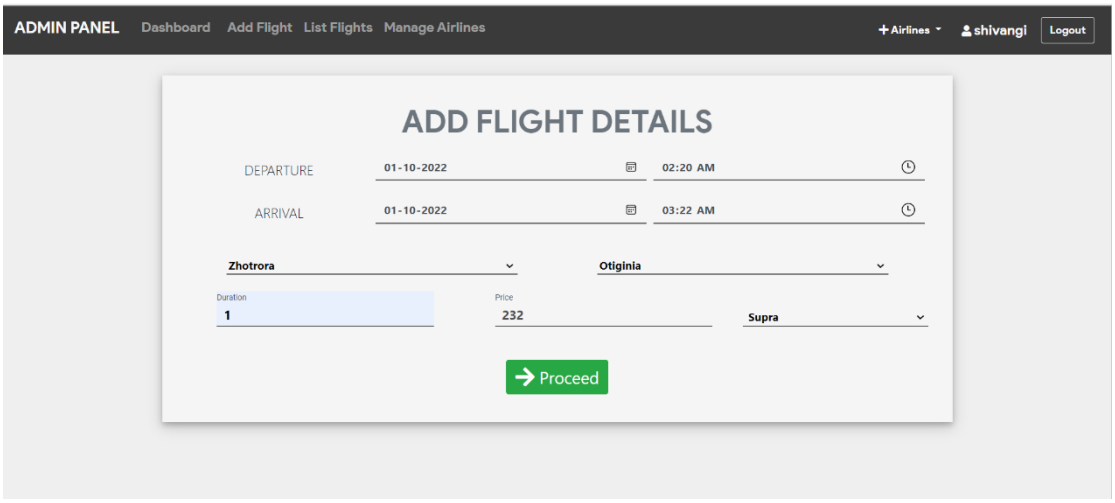
Admin Log In:

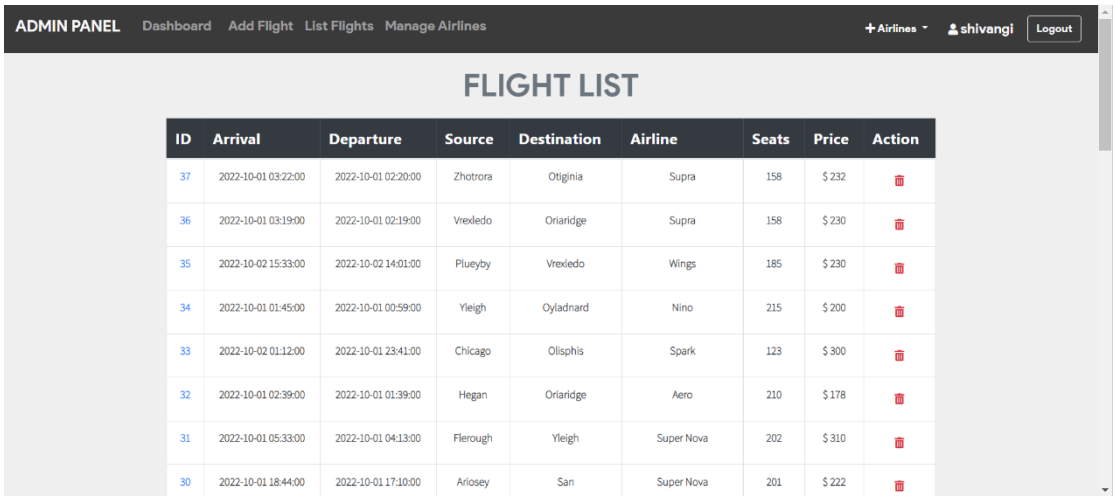


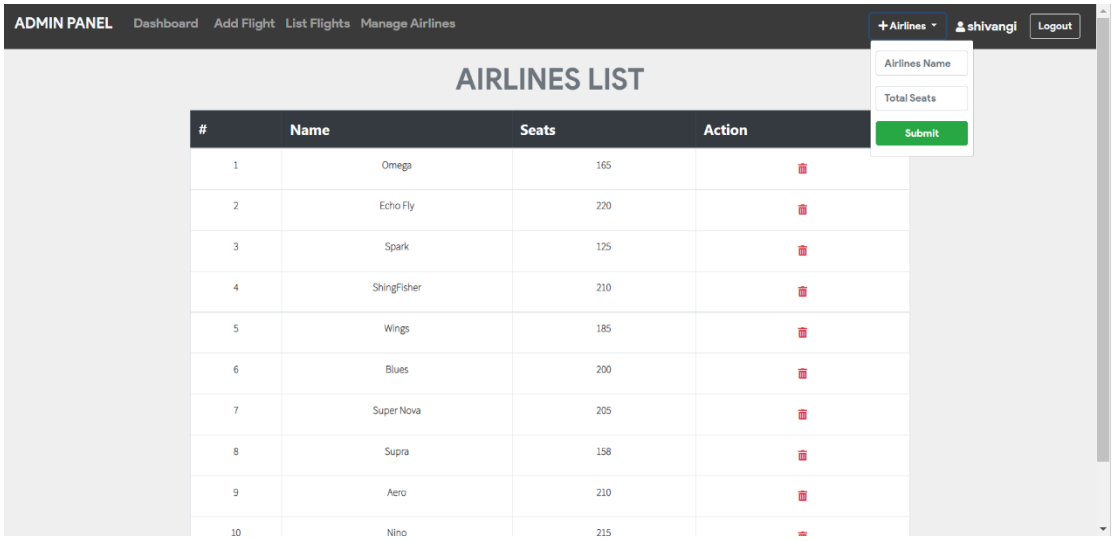
Add Flights:


ADMIN PANEL
Dashboard
Add Flight
List Flights
Manage Airlines
+Airlines
shivangi
Logout
ADD FLIGHT DETAILS
DEPARTURE
01-10-2022
02:20 AM
ARRIVAL
01-10-2022
03:22 AM
Zhotrora
Otiginia
Duration
1
Price
Select Airline
Please fill out this field.
Proceed

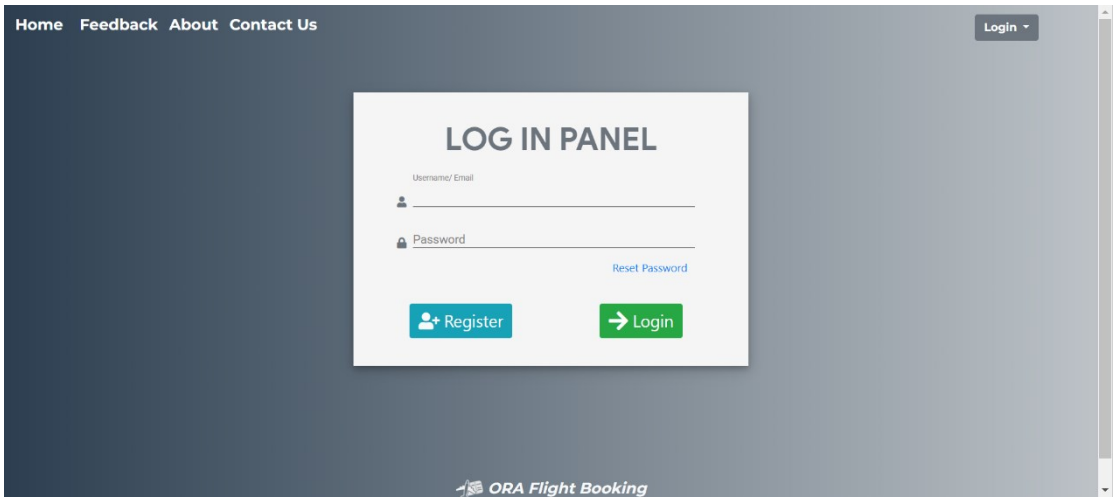

ADMIN PANEL
Dashboard
Add Flight
List Flights
Manage Airlines
+Airlines
shivangi
Logout
ADD FLIGHT DETAILS
DEPARTURE
01-10-2022
02:20 AM
ARRIVAL
01-10-2022
03:22 AM
Zhotrora
Otiginia
Duration
1
Price
232
Supra
Proceed


ADMIN PANEL
Dashboard
Add Flight
List Flights
Manage Airlines
+Airlines
shivangi
Logout
FLIGHT LIST
ID
Arrival
Departure
Source
Destination
Airline
Seats
Price
Action
37 2022-10-01 03:22:00 2022-10-01 02:20:00 Zhotrora Otiginia Supra 158 $ 232
36 2022-10-01 03:19:00 2022-10-01 02:19:00 Vrexledo Oriaridge Supra 158 $ 230
35 2022-10-02 15:33:00 2022-10-02 14:01:00 Plueyby Vrexledo Wings 185 $ 230
34 2022-10-01 01:45:00 2022-10-01 00:59:00 Yleigh Oyladnard Nino 215 $ 200
33 2022-10-02 01:12:00 2022-10-01 23:41:00 Chicago Olisphis Spark 123 $ 300
32 2022-10-01 02:39:00 2022-10-01 01:39:00 Hegan Oriaridge Aero 210 $ 178
31 2022-10-01 05:33:00 2022-10-01 04:13:00 Flerough Yleigh Super Nova 202 $ 310
30 2022-10-01 18:44:00 2022-10-01 17:10:00 Ariosey San Super Nova 201 $ 222

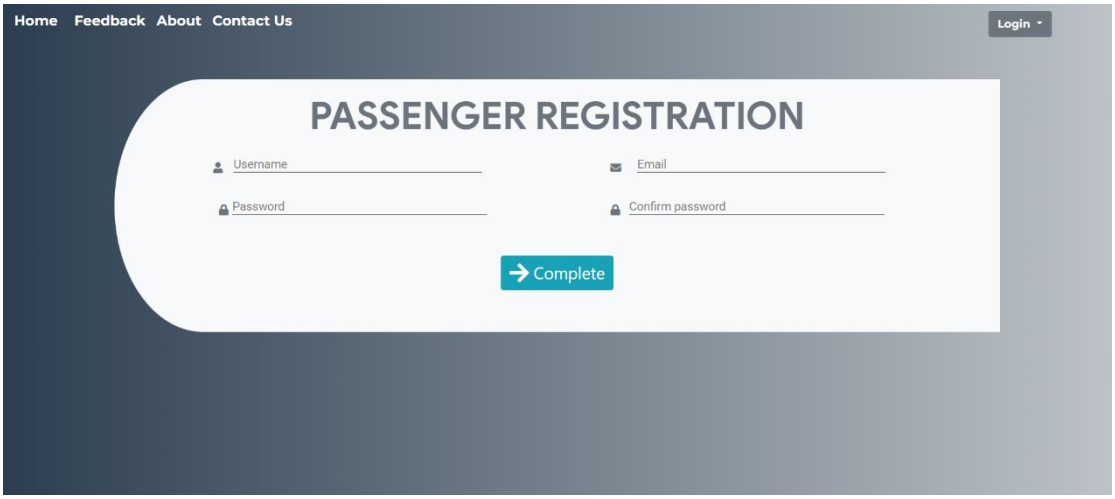
Add Airline :



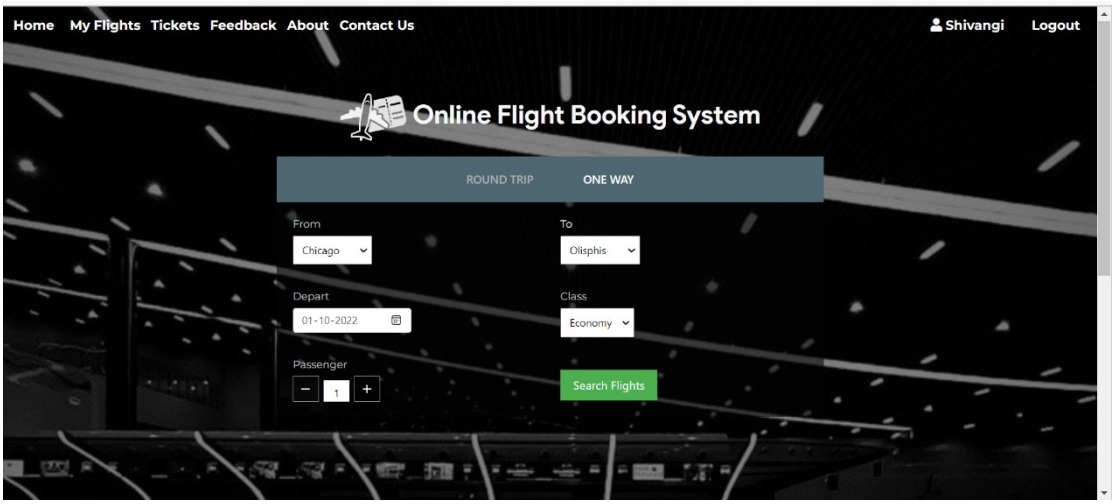

User Login:

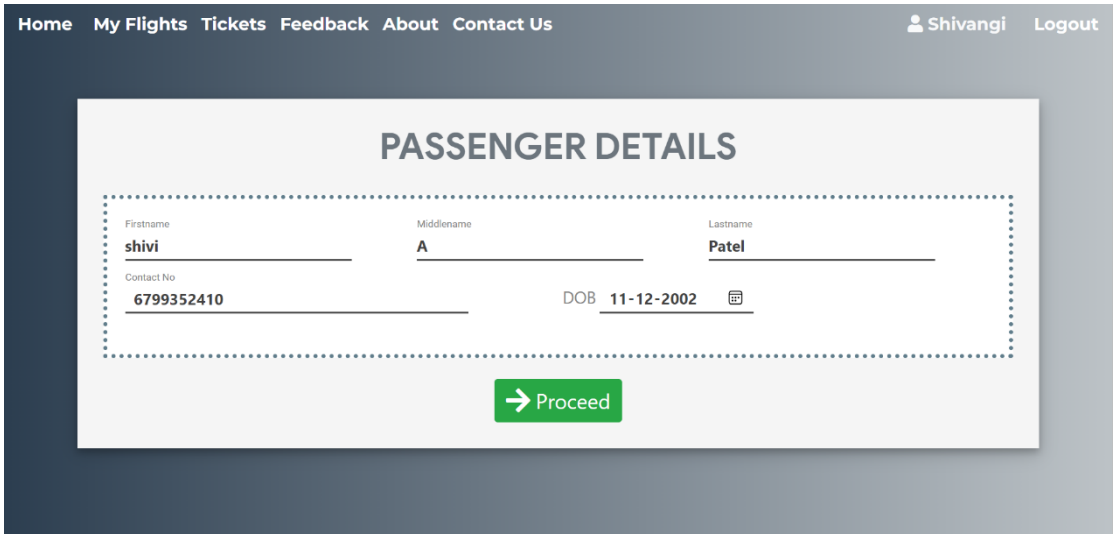


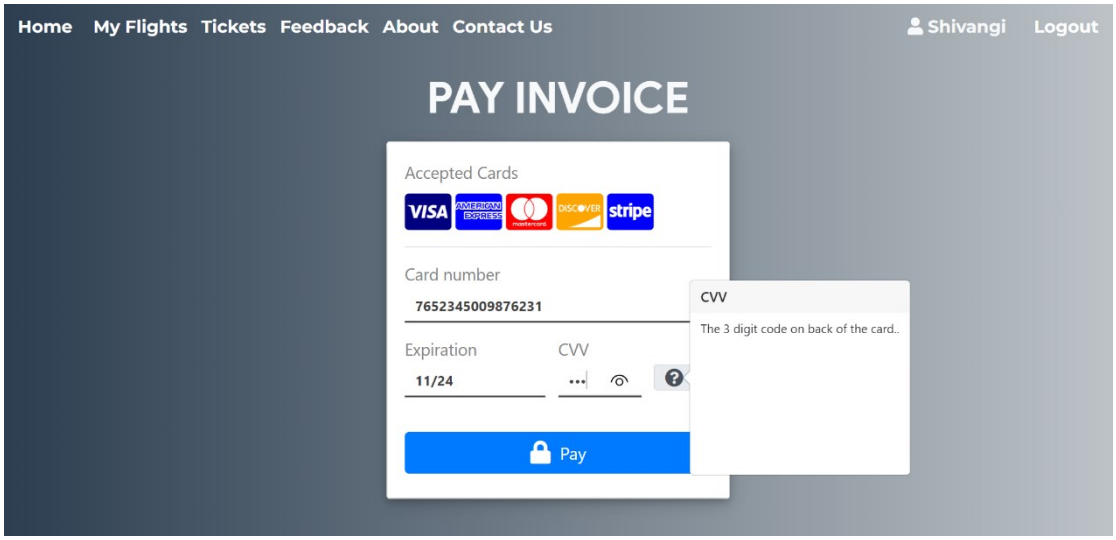
User Registration:

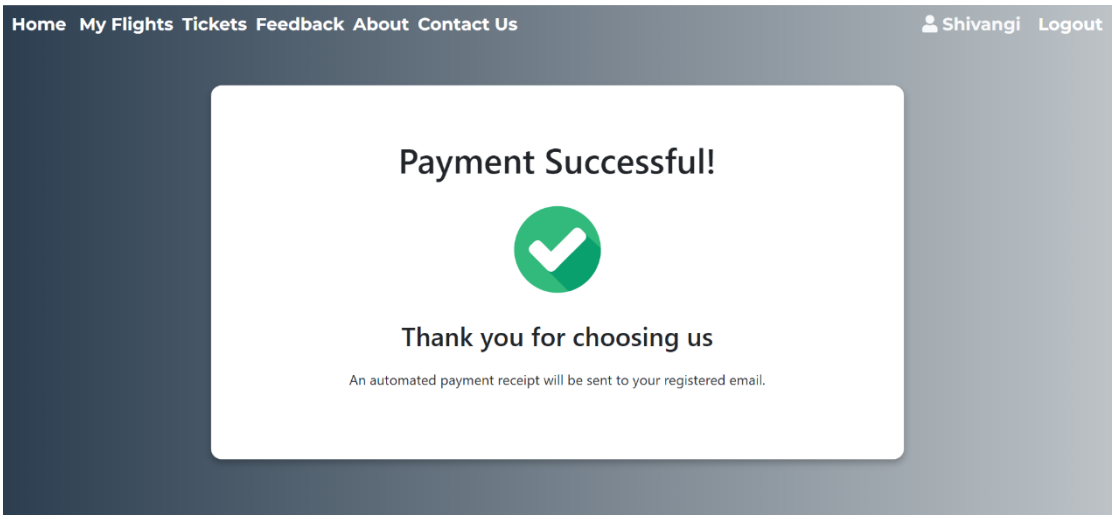


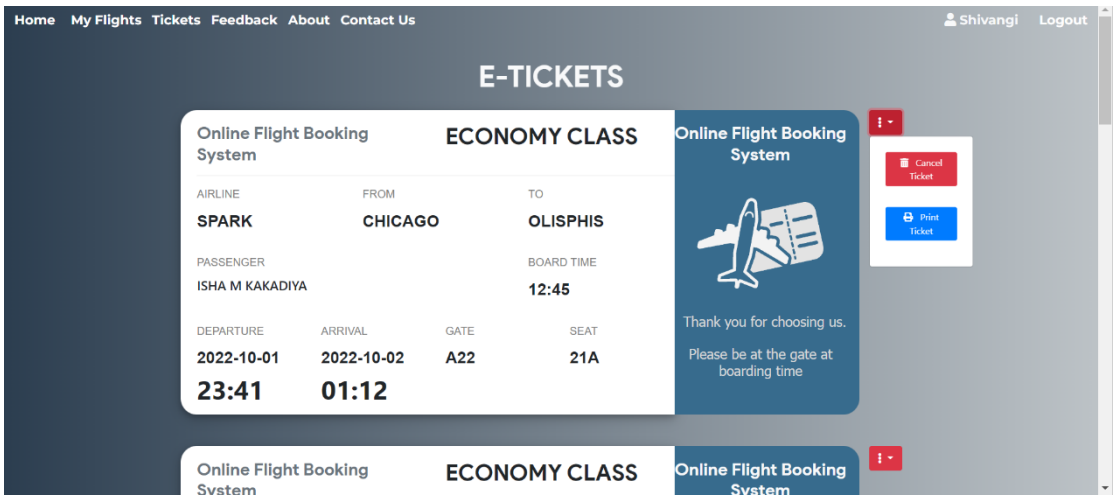

Book Flight:

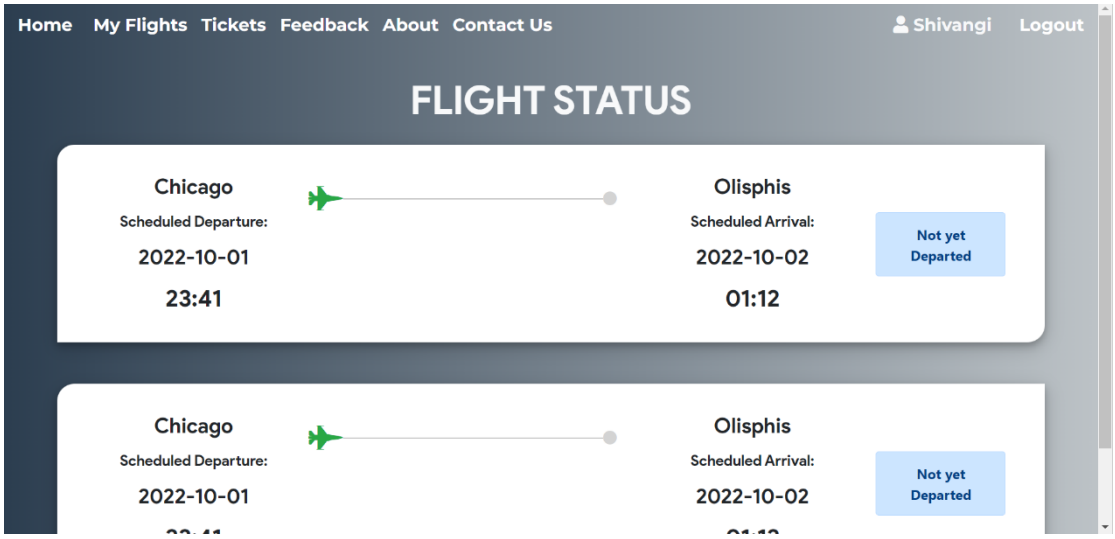



Home My Flights Tickets Feedback About Contact Us
Shivangi Logout
PASSENGER DETAILS
Firstname
shivi
Middlename
A
Lastname
Patel
Contact No
6799352410
DOB 11-12-2002
Proceed

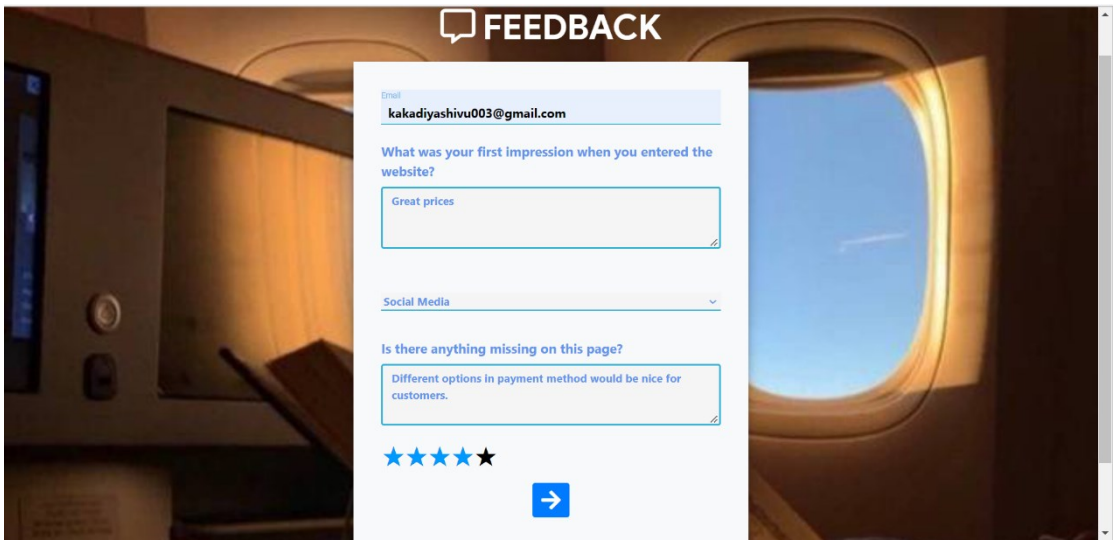

Home My Flights Tickets Feedback About Contact Us
Shivangi Logout
PAY INVOICE
Accepted Cards
Card number
7652345009876231
Expiration
11/24
CVV
Pay
CVV
The 3 digit code on back of the card..


Home My Flights Tickets Feedback About Contact Us
Shivangi Logout
Payment Successful!
Thank you for choosing us
An automated payment receipt will be sent to your registered email.

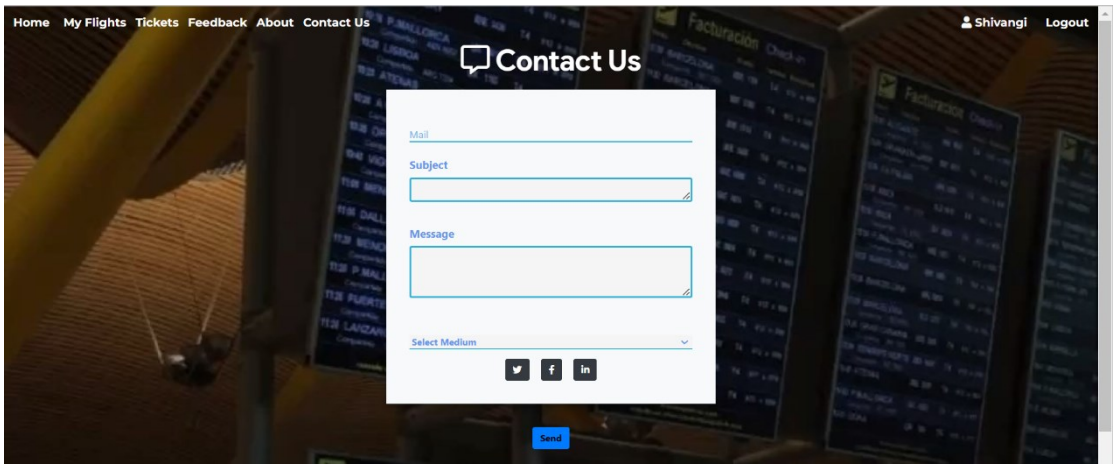



Give Feedback:

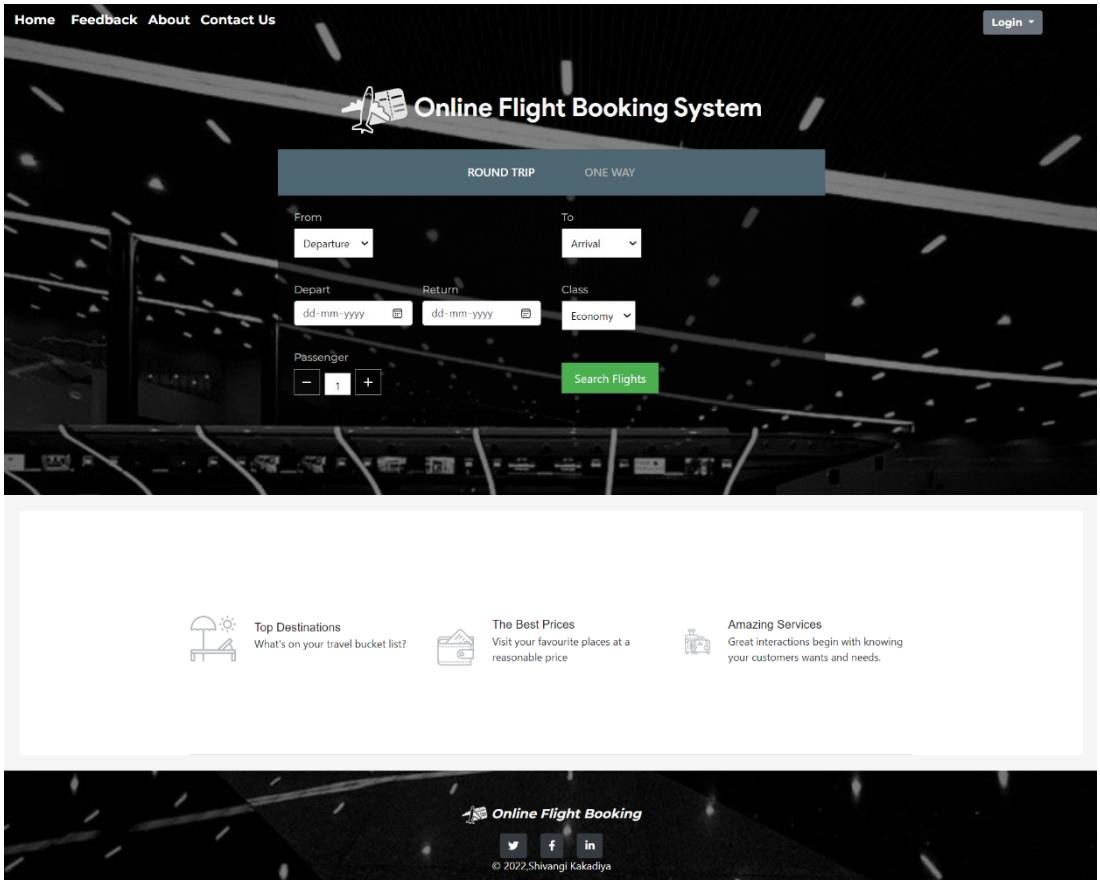


Contact :

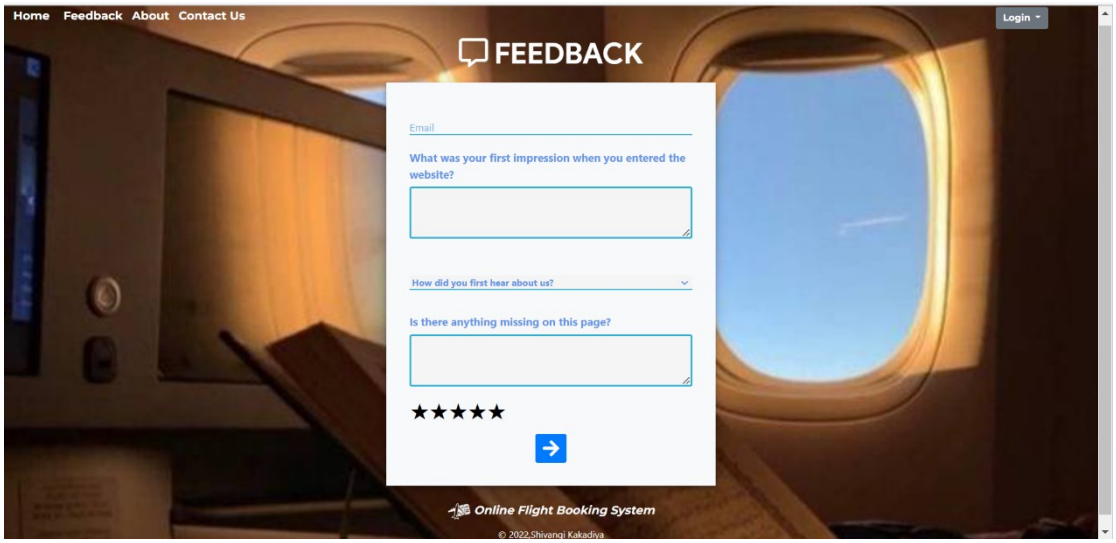



## 6.4 Output Design:

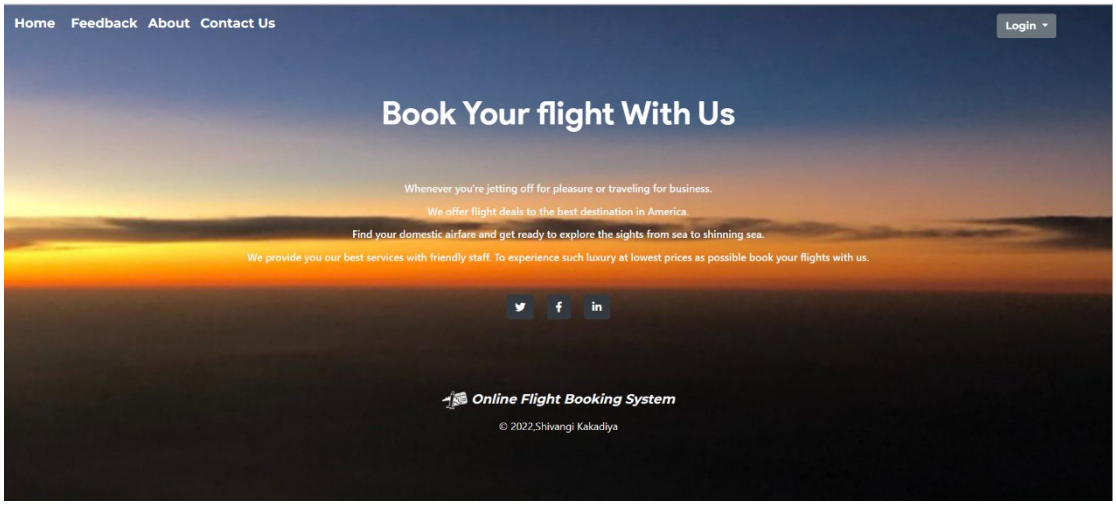
Home Page:



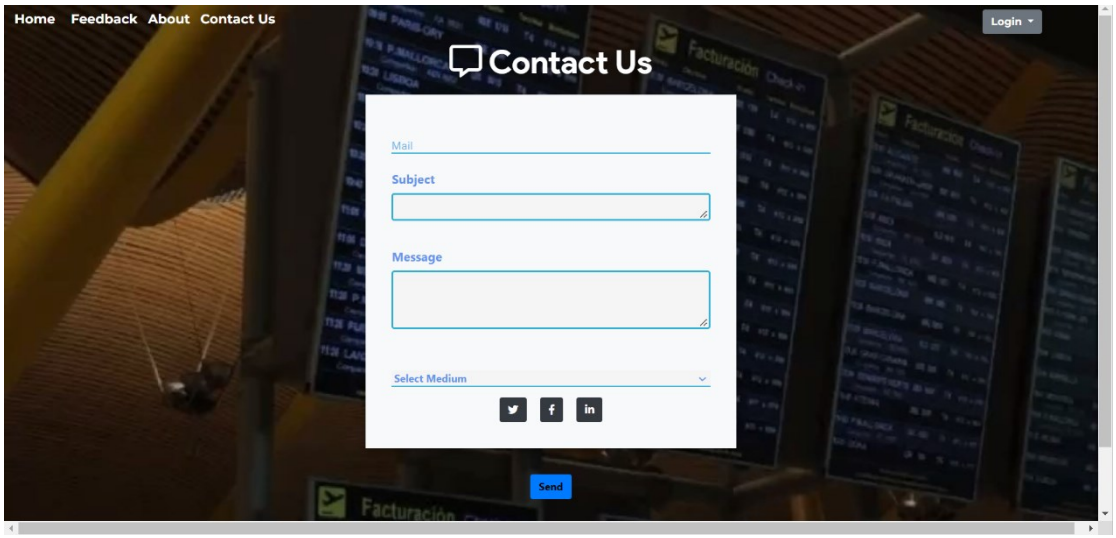
Feedback:




About page:




Contact us:



# 7. Software Testing

The testing process focuses on the logical intervals of the software ensuring that all statements have been tested and on functional interval is conducting tests to uncover errors and ensure that defined input will produce actual results that agree with the required results. Program level testing, modules level testing integrated and carried out.

Functional Testing:

- All web page is working properly.
- All navigation work properly.
- MySQL database work Proper.
- All Pages Design is perfect.

Environment Testing :

- Internet explorer and chrome consider testing for environment
- operability of software.
- Web server - IIS/Apache
- Database – SQL Server Management Studio
- OS – Windows 11
- Browser – Internet Explorer/Chrome

# 8.Limitations and Future Scope of Enhancements

**Limitations**:

Although I have put my best efforts to make the software flexible, easy to operate but limitations cannot be ruled out even by me. Though the software presents a broad range of options to its users some intricate options could not be covered into it; partly because of logistic and partly due to lack of sophistication. Paucity of time was also major constraint, thus it was not possible to make the software foolproof and dynamic.
Lack of time also compelled me to ignore some part such as storing old result of the candidate etc.

**Scope:**

The future scope includes expand the technologies like HTML and PHP we can also add new technologies like Laravel, reactjs many more for improving the efficiency of the software.

The Online Flight Booking system is the next generation address book which will provide these two basic services like portability, security.

The project will be useful for any schools and colleges with slightly modification. Project is flexible ie. any change /modification in database may be performing easily.

- This project can be upgraded by adding more options such as Ticket editing and more admin operations.
- Payment options and document checking such as ID proofs can be added.
- Applications can be upgraded by improving performance as per user feedback.

# 9. References

https://www.w3schools.com/php/php_sessions.asp
https://www.w3schools.com/Php/php_mysql_connect.asp
https://www.tutorialspoint.com/javascript/javascript_regexp_object.htm
https://www.tutorialspoint.com/javascript/javascript_animation.htm
https://www.codewithharry.com/videos/learn-php-in-one-video-in-hindi-2020/
https://www.movinnza.in/blog/payment-gateway-integration-php-ci/
http://talkerscode.com/webtricks/password-reset-system-using-php.php

Books:
PHP: A Beginner's Guide – by Vikram Vaswani

https://www.kobo.com/gr/en/ebook/php-a-beginner-s-guide-2