***Project Documentation:-***

*Contents:-*

//Project and Developer details

//GitHub Repository : https://github.com/Charul-javatsd/Phase2\_Ticket\_booking\_portal.git

***Project and Developer Details*:-**

* ***Developer Details:*** Charul Jain

***Project Details:***

* ***Project name :*FlyAway (An Airline Booking Portal).**

Flyaway is a ticket-booking portal that lets people book flights on their website.

* ***Project Description:***
* ● A search form in the homepage to allow entry of travel details, like the date of travel, source, destination, and the number of persons.
  + Based on the travel details entered, it will show the available flights with their ticket prices.
  + Once a person selects a flight to book, they will be taken to a register page where they must fill in their personal details. In the next page, they are shown the

flight details of the flight that they are booking, and the payment is done via a dummy payment gateway. On completion of the payment, they are shown a confirmation page with the details of the booking.

* For the above features to work, there will be an admin backend with the following features:
* ● An admin login page where the admin can change the password after login, if he wishes
* A master list of places for source and destination
* A master list of airlines
* A list of flights where each flight has a source, destination, airline, and ticket price

The goal of the company is to deliver a high-end quality product as early as possible.

* ***Tools used:***
  1. Eclipse IDE
  2. MySQL
  3. Apache Tom-Cat 10 server
  4. java servlets
  5. JSP
  6. HTML
  7. Apache Maven
  8. JDBC java Database connectivity

**Algorithm:**

**Step 1.1.1:** Creating a dynamic web project

* Open Eclipse
* **Go the File menu. Choose New->Dynamic Web Project**
* Enter the project name as FlyAway. Click on **Next**
* Enter nothing in the next screen and click on **Next**
* Check the checkbox **Generate web.xml deployment descriptor** and click on **Finish**
* This will create the project files in the Project Explorer
* Update Pom.xml

**Step 1.1.2:** Creating class

* In the Project Explorer, expand the project **FlyAway**
* Enter the Class Name as ConnectionUtil and click on **Finish**

**Step 1.1.3:** CreateServlets

* FlightInsert
* ForwardPassword
* List Of Flight
* Login Of Admin
* Login Of User
* Logout
* String Util

**Step 1.1.3:** CreateJSP

Admin Home

Admin Page

Booking Flight

Flight List

Forget Password

Home Page

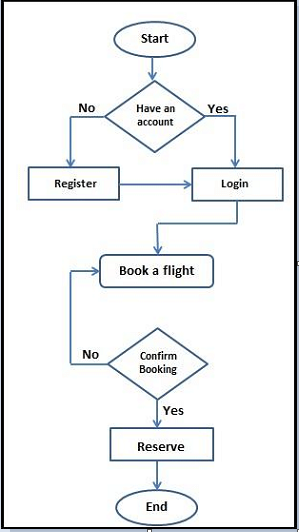
User Page

User Registration

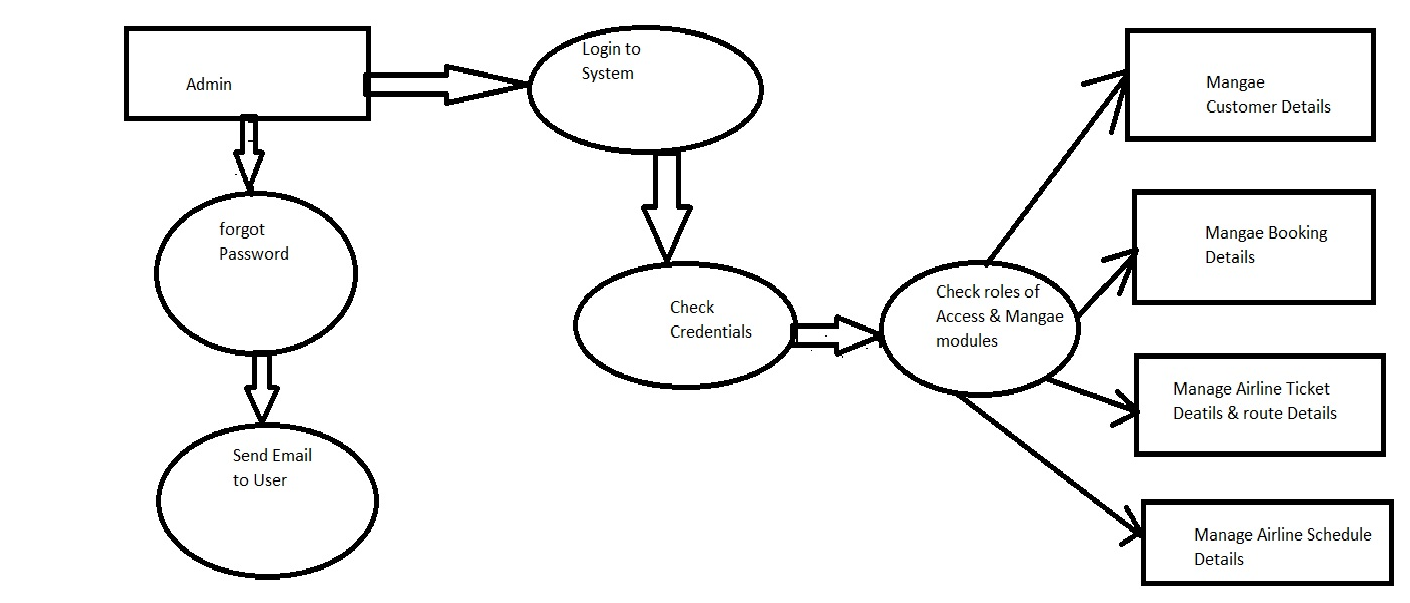
* ***Test Case: All Passed .***

**Flow Chart :**

**For User:**



**For Admin:**

****

**Sprint Plan:**



**Concepts used in this program:**

1. Object-Oriented: used to create and model objects for users and their credentials.
2. Databases: used to store and retrieve data.
3. Data Sources: used to define a set of properties required to identify and access the database.
4. Collections: used some collections such as List to store collection of data.
5. Exception Handling: used to catch problems that arises in the code especially in I/O blocks.
6. Cookies: to store log-in data on the client browser. Searching and Sorting

* ***GitHub Repository link:-*** ***https://github.com/Charul-javatsd/Phase2\_Ticket\_booking\_portal.git***