**FlyAway**

**Steps**

**STEP 1:** Create a database in MySQL named as a flyaway.

**STEP 2:** Create a table named user with values email, password, name, Phone

number, and Adhaar Number.

**STEP 3:** Create a table of flights with values from, to, date, time, and price.

**STEP 4:** Create a table admin and insert values admin mail id and password.

**STEP 5:** Create a maven project in eclipse named FlyAway.

**STEP 6:** Add dependencies in the pom.xml file.

**STEP 7:** Create a JSP file named Admin page for admin login.

**STEP 8:** Create a JSP file named as Admin Home page for admin usage

purposes.

**STEP 9:** Create a JSP file named Reset Password as the admin wish they can

reset the password.

**STEP 10:** Home Page JSP file is created for the user to login.

**STEP 11:** Insert Flight servlet will fetch the details entered by the admin and

add them to the MySQL database.

**STEP 12:** User Registration file which asks the user to register with personal

details. Once the user logged in it will ask for the flight details to

be booked. Flightlist.JSPwill display the available flights and the user

can select the flight and go for a payment. BookFlight.jspwill ask for the

payment and book the flight.

**STEP 13:** Book Flight jsp which asks the user to enter the payment details.

Once it is successful it will display Flight Booked successfully.

**STEP 14**: DAO class which connects the java with the database. And tables

and columns were fetched from the database.

**STEP 15:**StringUtil which handles the charactersandsymbols fetches from the

user and admin as well.

**STEP 16:** AdminLoginservlet will get the parameters entered by the user. If

the inputs are valid it takes them to the AdminHome. If it is invalid it

shows invalid details to the user.

**STEP 15:** ResetPasswordservlet gets the input from the admin and saves it to

The database.

**STEP 16:** UserRegistrationgets the input from the user and save it to the

database.

**STEP 17:** UserLoginservlet will get the inputs from the user and checks it with

the database and navigate to the next page.

**STEP 18:** The flight Listservlet will get the inputs from the user and checks it with

The database displays the flight if it is available or displays no flight

available if it is not in the database. If the flights are available it takes

the payment entered by the user.

**STEP 19:** Logoutservlet logs out and takes it to the homepage.

**STEP 20:** Run the project on the server, Tomcat 8.5.

**STEP 21:** Clone repository of git hub to the Eclipse.

**STEP 22:** Stage the file from unstage and commit the message.

**STEP 23:** Push the File Handling file in GitHub through Eclipse**.**

**STEP 24:**  The project is developed by the developer Heera Mani K B S.

**SPRINT PLANNING :**

**SPRINT 1:**

**SET UP THE WORKSPACE IN ECLIPSE IDE**

**CREATING GITHUB REPOSITORY TO KEEP**

**TRACKING**

**SET UP MAVEN PROJECT AND TOMCATE SERVER 8.5**

**FLOW OF APPLICATIONS USING SERVLETS,JSP MYSQL**

**SPRINT 2:**

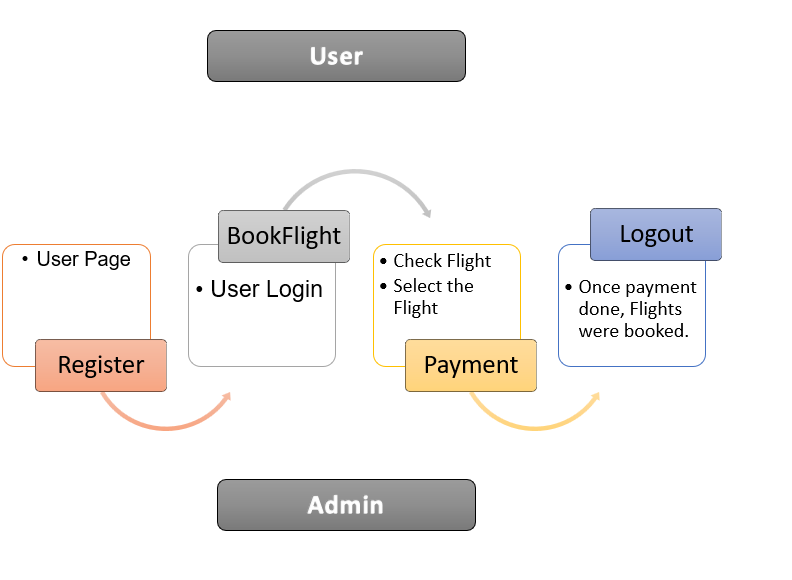
**JAVA PROGRAM TO IMPLEMENT ALL CONCEPTS**

**EXECUTION OF PROGRAM AS PER REQUIREMENTS RUNING THEM IN TOMCAT 8.5 SERVER**

**PUSH THE CODE TO GITHUB**

**CHECK THE COMPLETION STATUS OF PROJECT**

**FLOWCHART:**

****

.