INDEX

Chapter 1

INTRODUCTION

1.1 Project Overview

Flight delays could always be annoying, especially in the case when the period of delay was so long that there was even a danger to miss the next flight. However, if there was a way to predict whether there would be a delay or even better – how long the delay could be, then people could make earlier preparation to reschedule following flights in an earlier manner. For that consideration, we adopted a dataset containing airline delayed time to build a model, mainly aiming to solve the following questions. Whether there would be a delay with certain publicly reachable resources. How long delayed time one could expect with the same information given. We deployed python s k learn and pandas library to build our model, and evaluate our model based on R-Square for linear regression and accuracy rate for logistic regression. As a brief result of our project, we found, it would be helpful to use the following factors in evaluating our model: week, month, airline carrier reference, planned elapsed time (in air time), distance between two departure and destinations, flight planned departure time, departure airport code.

1.2 Purpose

As people increasingly choose to travel by air, the amount of flights that fail to take off on time also increases. This growth exacerbates the crowded situation at airports and causes financial difficulties within the airline industry. Air transportation delay indicates the lack of efficiency of the aviation system. It is a high cost to both airline companies and their passengers. According to the estimation by the Total Delay Impact Study, the total cost of air transportation delay to air travellers and the airline industry in 2007 was $32.9 billion in the US, resulting in a $4 billion reduction in GDP. Therefore, predicting flight delays can improve airline operations and passenger satisfaction, which will result in a positive impact on the economy. In this project, the main goal is to compare the performance of machine learning classification algorithms when predicting flight delays.

Chapter 2

LITERATURE REVIEW

2.1 Existing Problem

Flight delay creates a major problems in the current aviation systems and in scgeduling of airport operation,the unreliability of flight arrivals is a serious challenge.

Punctuality is an issue for all major careers, with some struggling more than others.

2.2 References

1. Flight Delay Prediction Based on Aviation Big Data and Machine Learning

Authors: Guan Gui, Fan Liu, Jinlong Sun, Jie Yang, Ziqi Zhou

Date of Publication: 18 November 2019

Accurate flight delay prediction is fundamental to establish the more efficient airline business. Recent studies have been focused on applying machine learning methods to predict the flight delay. Most of the previous prediction methods are conducted in a single route or airport. This paper explores a broader scope of factors which may potentially influence the flight delay, and compares several machine learning-based models in designed generalized flight delay prediction tasks. To build a dataset for the proposed scheme, automatic dependent surveillance-broadcast (ADS-B) messages are received, pre-processed, and integrated with other information such as weather condition, flight schedule, and airport information. The designed prediction tasks contain different classification tasks and a regression task. Experimental results show that long short-term memory (LSTM) is capable of handling the obtained aviation sequence data, but overfitting problem occurs in our limited dataset. Compared with the previous schemes, the proposed random forest-based model can obtain higher prediction accuracy (90.2% for the binary classification) and can overcome the overfitting problem

2. A deep learning approach to flight delay prediction

Authors: Young Jin Kim, Sun Choi, Simon Briceno, Dimitri Mavris

Date of publication: 12 December 2016

Deep learning has achieved significant improvement in various machine learning tasks including image recognition, speech recognition, machine translation and etc. Inspired by the huge success of the paradigm, there have been lots of tries to apply deep learning algorithms to data analytics problems with big data including traffic flow prediction. However, there has been no attempt to apply the deep learning algorithms to the analysis of air traffic data. This paper investigates the effectiveness of the deep learning models in the air traffic delay prediction tasks. By combining multiple models based on the deep learning paradigm, an accurate and robust prediction model has been built which enables an elaborate analysis of the patterns in air traffic delays. In particular, Recurrent Neural Networks (RNN) has shown its great accuracy in modeling sequential data. Day-to-day sequences of the departure and arrival flight delays of an individual airport have been modeled by the Long Short-Term Memory RNN architecture. It has been shown that the accuracy of RNN improves with deeper architectures. In this study, four different ways of building deep RNN architecture are also discussed. Finally, the accuracy of the proposed prediction model was measured, analyzed and compared with previous prediction methods. It shows best accuracy compared with all **other** methods.

3. Airline delay prediction by machine learning algorithms

Authors: H. Khaksar, A. Sheikholeslami

Date of publication: 28 January 2017

Flight planning, as one of the challenging issue in the industrial world, is faced with many uncertain conditions. One such condition is delay occurrence, which stems from various factors and imposes considerable costs on airlines, operators, and travelers. With these considerations in mind, we implemented flight delay prediction through proposed approaches that are based on machine learning algorithms. Parameters that enable the effective estimation of delay are identified, after which Bayesian modeling, decision tree, cluster classification, random forest, and hybrid method are applied to estimate the occurrences and magnitude of delay in a network. These methods were tested on a U.S. flight dataset and then refined for a large Iranian airline network. Results showed that the parameters affecting delay in US networks are visibility, wind, and departure time, whereas those affecting delay in Iranian airline flights are fleet age and aircraft type. The proposed approaches exhibited an accuracy of more than 70% in calculating delay occurance and magnitude in both the whole-network US and Iranian. It is hoped that the techniques put forward in this work will enable airline companies to accurately predict delays, improve flight planning, and prevent delay propagation.

4. Prediction of weather-induced airline delays based on machine learning algorithms

Authors: Sun Choi, Young Jin Kim, Simon Briceno, Dimitri Mavris

Date of publication: 12 December 2016

The primary goal of the model proposed in this paper is to predict airline delays caused by inclement weather conditions using data mining and supervised machine learning algorithms. US domestic flight data and the weather data from 2005 to 2015 were extracted and used to train the model. To overcome the effects of imbalanced training data, sampling techniques are applied. Decision trees, random forest, the AdaBoost and the k-Nearest-Neighbors were implemented to build models which can predict delays of individual flights. Then, each of the algorithms' prediction accuracy and the receiver operating characteristic (ROC) curve were compared. In the prediction step, flight schedule and weather forecast were gathered and fed into the model. Using those data, the trained model performed a binary classification to predict whether a scheduled flight will be delayed or on-time.

2.3 Problem statement definition

Flight Planning is one of the challenges in industrial world which faces many uncertain conditions. One such condition is delay occurrence, which stems from various factors and imposes considerable cost so airlines, operators, and travellers. Delays in departure can occur due to bad weather conditions, seasonal and holiday demands, airline policies, technical issue such as problems in airport facilities, luggage handling and mechanical apparatus, and accumulation of delays from preceding flights. Here in flight delay prediction system based on the weather parameters which can result in delays. The system considers the temperature, humidity, rain in mm, visibility and month number as important parameters for prediction of delay.

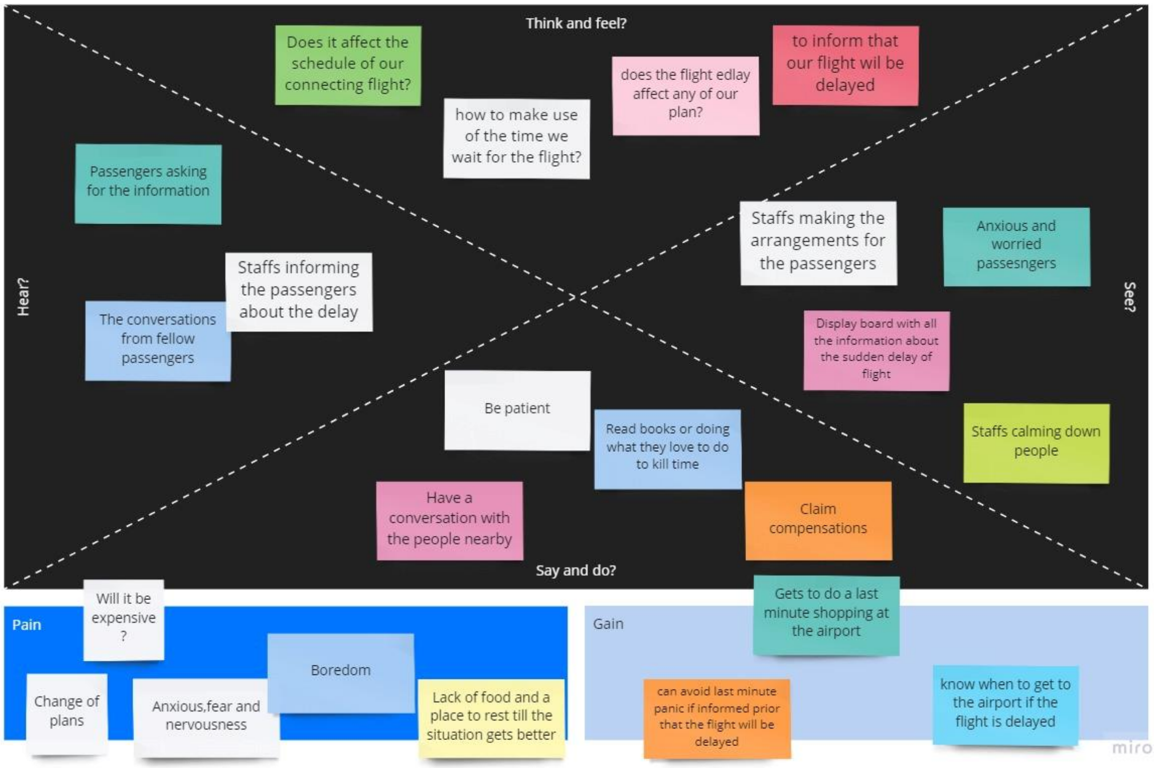
Major problem which causes delay in flights can be delay propagation, delay caused on the departure point or the root of the flight, and cancellation of flights. These problems cannot be eliminated forever, but a delay prediction tool will allow the operator and the administrators to take the concerned actions for smooth operation. Various methods that can be used to develop a system which predicts the delay in flights can be Machine Learning, Probabilistic models, Statistical analysis or Network Representations.

Chapter 3

IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas

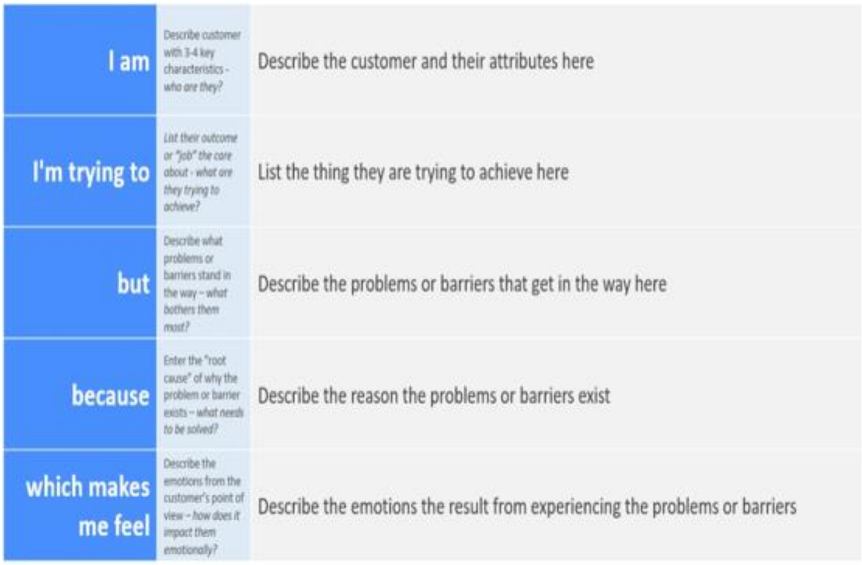
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.


Think and feel?
Does it affect the schedule of our connecting flight?
how to make use of the time we wait for the flight?
does the flight edlay affect any of our plan?
to inform that our flight will be delayed
Hear?
Passengers asking for the information
Staffs informing the passengers about the delay
The conversations from fellow passengers
See?
Staffs making the arrangements for the passengers
Anxious and worried passesngers
Display board with all the information about the sudden delay of flight
Staffs calming down people
Say and do?
Be patient
Read books or doing what they love to do to kill time
Have a conversation with the people nearby
Claim compensations
Pain
Will it be expensive ?
Change of plans
Anxious,fear and nervousness
Boredom
Lack of food and a place to rest till the situation gets better
Gain
Gets to do a last minute shopping at the airport
can avoid last minute panic if informed prior that the flight will be delayed
know when to get to the airport if the flight is delayed
miro

3.2 Ideation & Brainstorming

Ideation is the creative process of generating, developing, and communicating new ideas, where an idea is understood as a basic element of thought that can be either visual, concrete, or abstract. Ideation comprises all stages of a thought cycle, from innovation, to development, to actualization.

In our project we have researched about different kinds of people and the problems they might face if they face the flight delay issue and also how they might feel knowing their schedule might be changed because of the delay



I am	*Describe customer with 3-4 key characteristics - who are they?*	Describe the customer and their attributes here
I'm trying to	*List their outcome or "job" the care about - what are they trying to achieve?*	List the thing they are trying to achieve here
but	*Describe what problems or barriers stand in the way – what bothers them most?*	Describe the problems or barriers that get in the way here
because	*Enter the "root cause" of why the problem or barrier exists – what needs to be solved?*	Describe the reason the problems or barriers exist
which makes me feel	*Describe the emotions from the customer's point of view – how does it impact them emotionally?*	Describe the emotions the result from experiencing the problems or barriers

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	A Business Man	Fly to Delhi to attend a conference	Air craft is delayed	Preparing the aircraft	Drive up the wall
PS-2	Doctor	Fly to London for an operation	Delay in reaching the destination	Adverse weather condition	Anxious

PS-3	Singer	Fly to Amsterdam fora concert	Delay in landing	Aircraft traffic control restrictions	Dejected

Brainstorming:

Brainstorming is a situation where a group of people meet to generate new ideas and solutions around a specific domain of interest by removing inhibitions. People are able to think more freely and they suggest as many spontaneous new ideas as possible. All the ideas are noted down without criticism and after the brainstorming session the ideas are evaluated.

Key rules of brainstorming

- Stay in topic
- Defer judgement
- Go for volume
- Encourage wild ideas
- Listen to others

- If possible, be virtual

Step 1: Define your problem statement

The problem is to identify the cause of flight delay and to predict its future delays.

Step 2: Brainstorm

Write down any ideas that come to mind that address your problem statement. After analysing the each and every situation and possibility of our problem,these are all the ideas of our team members

Team member 1: Mariya Anthony Visuvasammal.S

The data can be collected from the internet and pre-processing step should be done. Machine learning algorithm can be performed to the pre-processed data. The model should be developed and webpage must be created. We can also ask for flight details from the clients. Using the model, the website will show the prediction. Passenger can plan accordingly.

Team member 2: Darsheta D

The model is built using the machine learning algorithm. The dataset containing the flight details is collected. Using weather condition, the model will built. The delays can be predicted using the model. A web page can be created for user interaction. Passengers will get to know about the delay. We can reduced the customer dissatisfaction. Passengers can predict the delay

Team member 3: Mary Rajam A

The reason for the flight delay should be predicted. The details about how to gte a refund or flight compensation is shown. Knowing the delays improve the customer trust. The dataset can be collected and machine learning algorithm should be performed. The results are analysed and using effective model, the prediction is done. By creating a webpage, where user can check the delays. The flight details are displayed

Team member 4: Anitha R

The model should be built. The reason why the flight is getting delayed is analysed. The reasons can be, airplane cleaning, equipment malfunction, closed runaway, bad weather. The question should be asked. Whether the information available on the cause of delays and cancellations? Deliver the delay reason to the passenger. Using machine learning algorithm the data is pre-processed and the webpage is created

3.3 PROPOSED SOLUTION:

Idea / Solution description:

Developing a model which gives accurate solution for flight delay with the help of web application and machine learning algorithms.

Novelty / Uniqueness:

The purpose of this project is to look at the approaches used to build models for predicting flight delays. using the attribute from the data set that beneficence in the prediction. Breaking all our variables into their sub categories.

Social Impact / Customer Satisfaction:

Flight delays have negative impacts on several aspects, such as passengers, airlines, and air transport systems. Delayed flights throw travel plans into disarray, often making passengers dissatisfied with the airlines.

Business Model (Revenue Model):

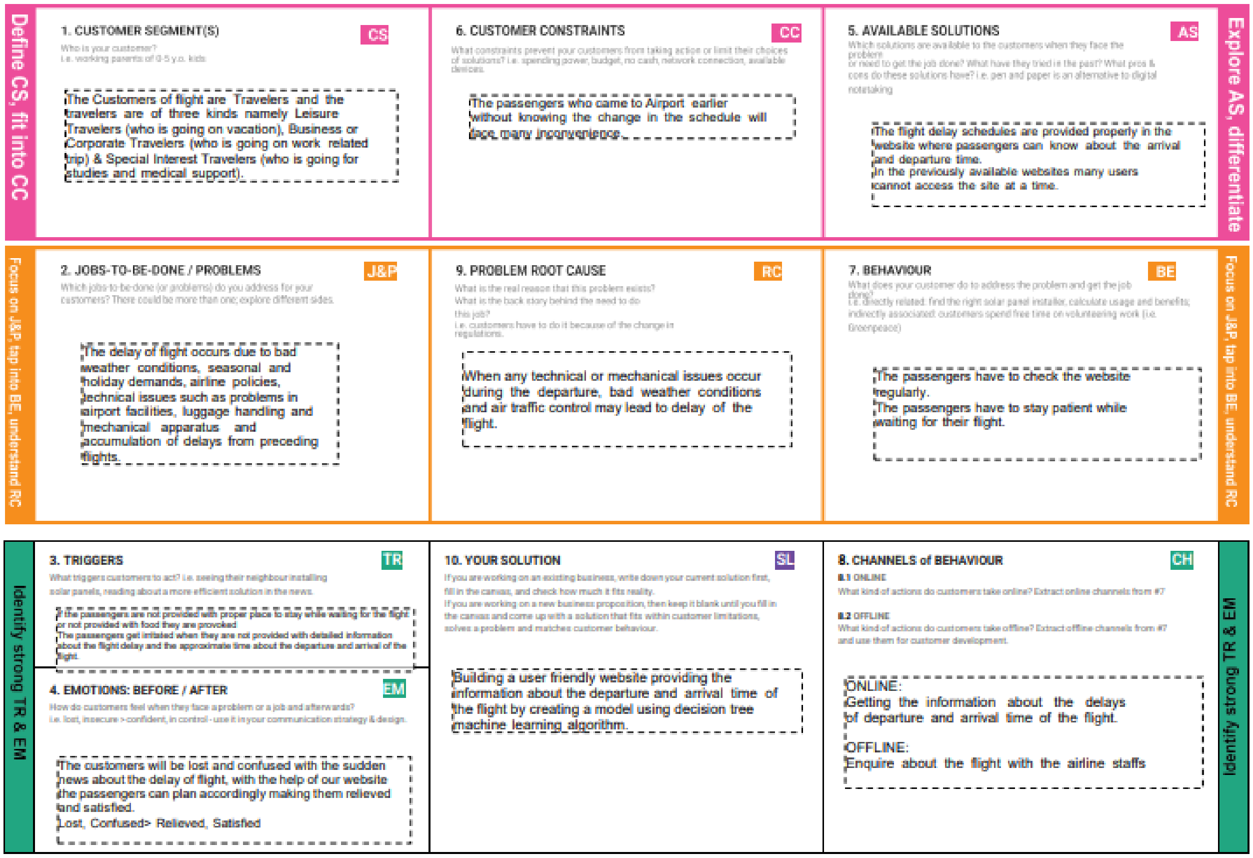
The solution is the low-cost airline model planned to be created as an application with which the consumer can interact directly to know the details of the flight.

Scalability of the Solution:

Identifying propagated departure delays and measuring their contribution to arrival delays. This makes the passenger to take preventive action when the status of the flight delay.

3.4 PROBLEM SOLUTION FIT:

Problem-Solution Fit - this occurs when you have evidence that customers care about certain jobs, pains, and gains. At this stage you've proved the existence of a problem and have designed a value proposition that addresses your customers' jobs, pains and gains.



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

The Customers of flight are Travelers and the travelers are of three kinds namely Leisure Travelers (who is going on vacation), Business or Corporate Travelers (who is going on work related trip) & Special Interest Travelers (who is going for studies and medical support).

6. CUSTOMER CONSTRAINTS CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

The passengers who came to Airport earlier without knowing the change in the schedule will face many inconvenience.

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem
or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

The flight delay schedules are provided properly in the website where passengers can know about the arrival and departure time.
In the previously available websites many users cannot access the site at a time.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

The delay of flight occurs due to bad weather conditions, seasonal and holiday demands, airline policies, technical issues such as problems in airport facilities, luggage handling and mechanical apparatus and accumulation of delays from preceding flights.

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

When any technical or mechanical issues occur during the departure, bad weather conditions and air traffic control may lead to delay of the flight.

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

The passengers have to check the website regularly.
The passengers have to stay patient while waiting for their flight.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

If the passengers are not provided with proper place to stay while waiting for the flight or not provided with food they are provoked.
The passengers get irritated when they are not provided with detailed information about the flight delay and the approximate time about the departure and arrival of the flight.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

The customers will be lost and confused with the sudden news about the delay of flight, with the help of our website the passengers can plan accordingly making them relieved and satisfied.
Lost, Confused> Relieved, Satisfied

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

Building a user friendly website providing the information about the departure and arrival time of the flight by creating a model using decision tree machine learning algorithm.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

ONLINE:
Getting the information about the delays of departure and arrival time of the flight.

OFFLINE:
Enquire about the flight with the airline staffs

Identify strong TR & EM

Chapter 4

REQUIREMENT ANALYSIS

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.

4.1 FUNCTIONAL REQUIREMENT:

Functional requirements are the desired operations of a program, or system as defined in software development and systems engineering. The systems in systems engineering can be either software electronic hardware or combination software-driven electronics. Following are the functional requirements of the proposed solution

User Registration:

Registration through Form Registration through Gmail

User Confirmation:

Confirmation via Email Confirmation via OTP

User Login:

Login using credentials

User Requirement:

Collecting information like date of voyage, departing & arrival destination, flight number and storing that information in the cloud.

Flight Status Notification:

The users can access and view the flight status at any moment.

Feedback:

The user's feedback information can be used to provide personalized recommendations.

4.2 NON-FUNCTIONAL REQUIREMENT:

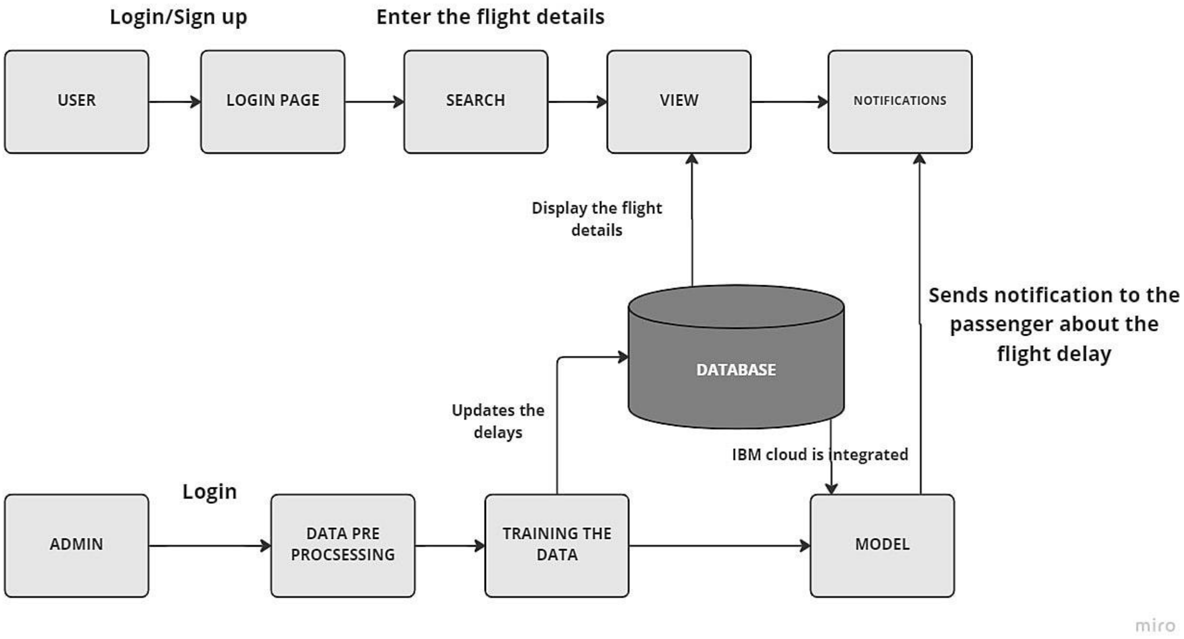
In systems engineering and requirements engineering, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. They are contrasted with functional requirements that define specific behavior or functions.

Chapter 5

PROJECT DESIGN

5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture:

A solution architecture (SA) is an architectural description of a specific solution. SAs combine guidance from different enterprise architecture viewpoints (business, information and technical), as well as from the enterprise solution architecture (ESA).

Technical Architecture (TA) is a form of IT architecture that is used to design computer systems. It involves the development of a technical blueprint with regard to the arrangement, interaction, and interdependence of all elements so that system-relevant requirements are met.

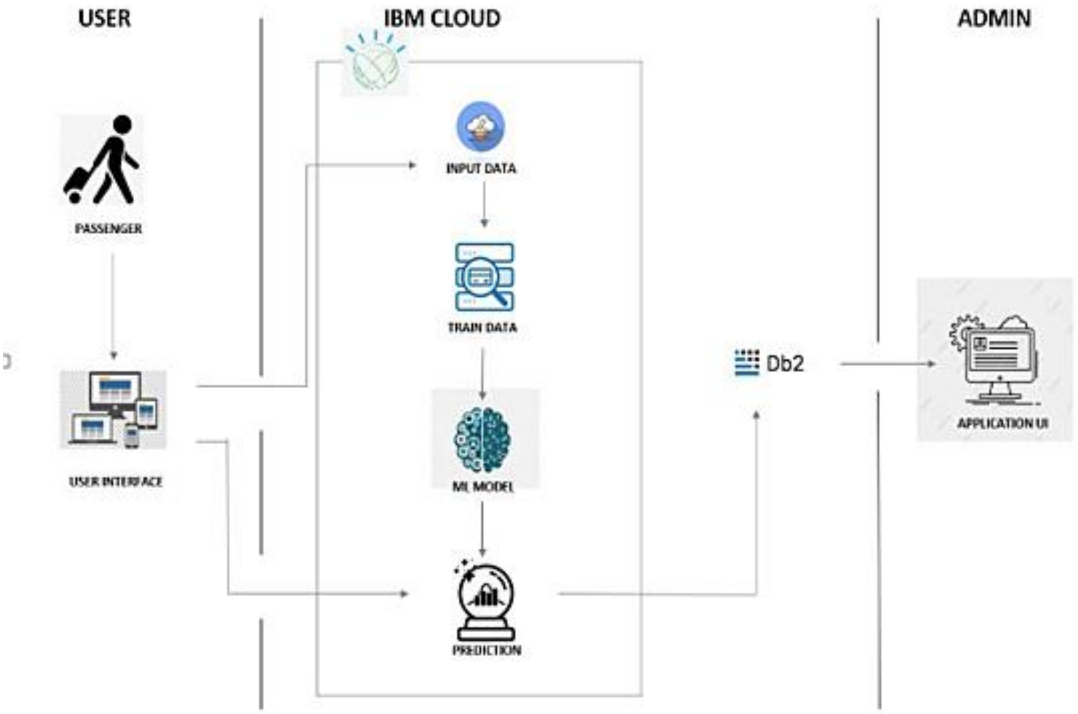
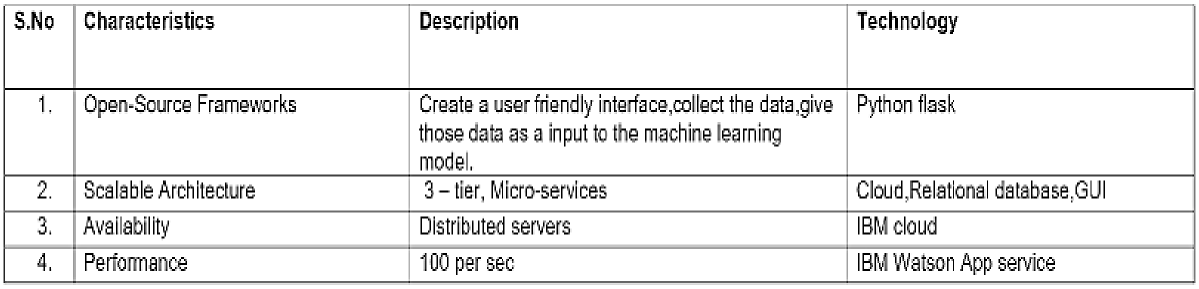


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	UI provides interaction between users and web application	HTML, CSS, JAVA SCRIPT.
2.	Application Logic-1	User can enter the details about the flight to know the delay	Python
3.	Application Logic-2	Application is directly deployed in the IBM cloud	IBM Watson STT service
4.	Database	In the Database,user details are stored and maintained for verifications.	MySQL
5.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
6.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
7.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
8.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

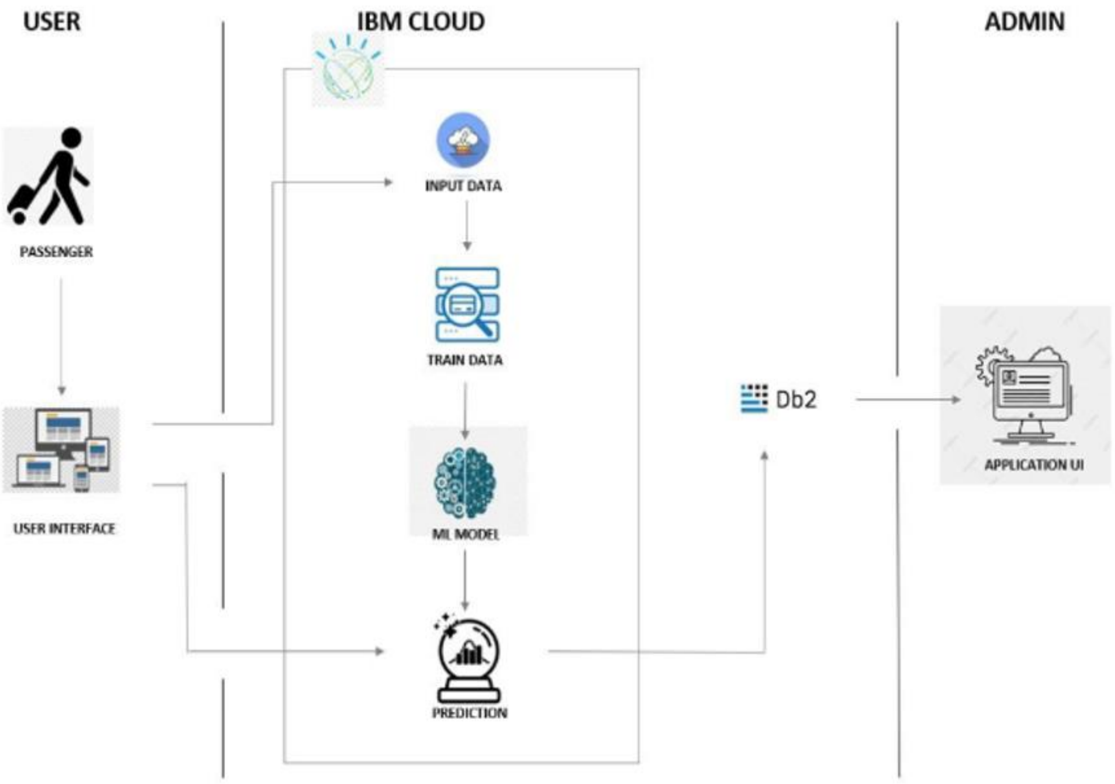
Table-2: Application Characteristics:



S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Create a user friendly interface,collect the data,give those data as a input to the machine learning model.	Python flask
2.	Scalable Architecture	3 – tier, Micro-services	Cloud,Relational database,GUI
3.	Availability	Distributed servers	IBM cloud
4.	Performance	100 per sec	IBM Watson App service

5.3 User Stories:

Use the below template to list all the user stories for the product



CHAPTER-6

PROJECT PLANNING & SCHEDULING:

6.1 Sprint Planning & Estimation:

Sprint 1:

- The functions which take place in this sprint 1 are Data Collection, Data Pre-Processing, Model Building and Model Evaluation.

- Data Collection and Data Pre-Processing are done so that the user can know about the attributes of data by presentation of the developer.

- Model Building is done so that the user can predict flight delay using various machine learning model.

- Model Evaluation is done so that the user can predict flight delay using efficient machine learning model.

Sprint 2:

- The functional requirements that take place during this sprint 2 are Model Deployment on IBM Cloud using IBM Watson and User Interaction Dashboard.

- The user can request the cloud to use the model by Model Deployment on IBM Cloud using IBM Watson.

- The user can interact with the dashboard to use and predict the model with the help of User Interaction Dashboard.

Sprint 3:

- The functional requirements that are done in this sprint 3 are Login and Registration.

- By the Registration function the user can register the application by using email and password.

- With the help of Login function the use can log into the application by using email and password.

Sprint 4:

- The functional requirements attained during this sprint 4 are Raising Query & Complaint and Improve overall web application.

- The user can raise the complain and give the feedback for the application during the function of Raising Query and Complaint.

- The user can use the revised version of web application during the Improve Overall Web Application of function is accomplished.

6.2 Sprint Delivery Schedule:

1. The Sprint 1 is estimated to be completed in six days duration, the Sprint 1 Start Date is on 24th Oct 2022 and the End Date is on 29th Oct 2022.

2. The Sprint 2 is estimated to be completed in six days duration, the Sprint 2 Start date is on 31st Oct 2022 and the End date is on 5th Nov 2022.

3. The Sprint 3 is estimated to be completed in six days duration, the Sprint 3 Start Date is on 7th Nov 2022 and the End Date is on 12th Nov 2022.

4. The Sprint 4 is estimated to be completed in six days duration, the Sprint 4 Start Date is on 14th Nov 2022 and the End Date is on 19th Nov 2022.

6.3 Reports from JIRA:

Departure & Arrival Delay Distribution

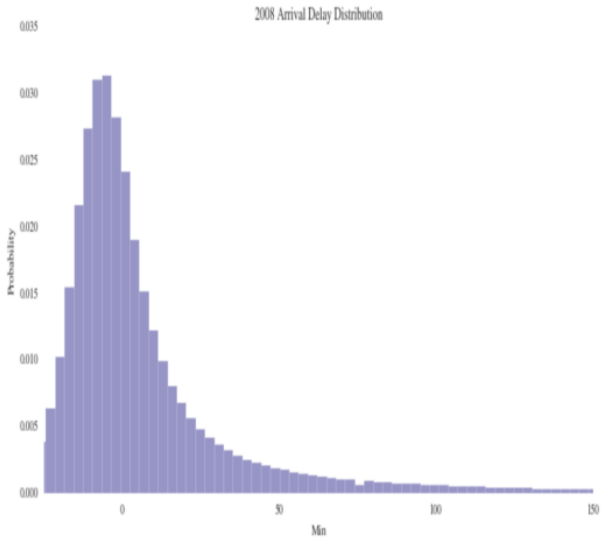


Fig6.3.1 ArrivalDistribution

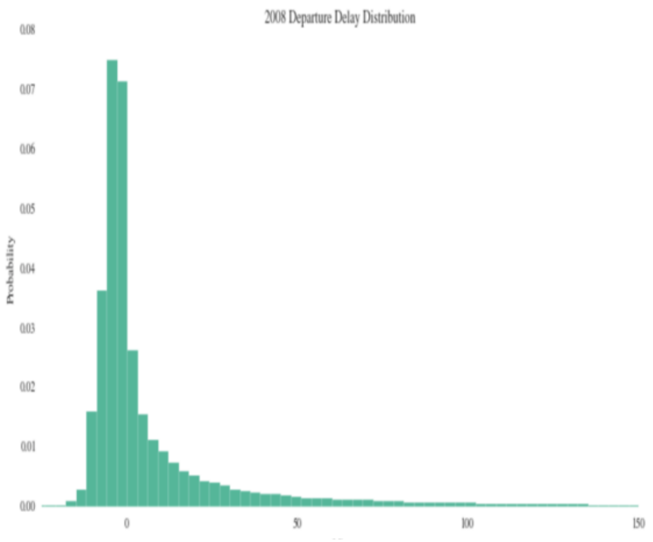


Fig 6.3.2 Delay Prediction

Average Departure & Arrival by Month

Next, we consider the impact of the months on the delays. We would expect that winter months have the most extended delay. A column chart with departure and arrival delay in minutes plotted by month is the most effective way to see the potential effects of the months.

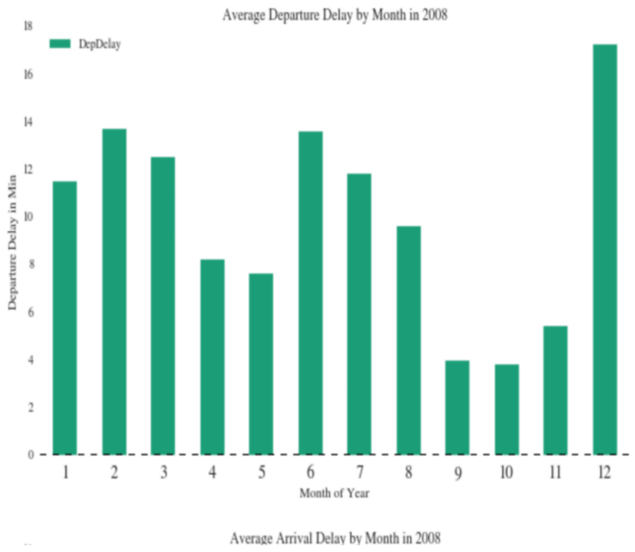


Fig 6.3.3 Average Departure Distribution

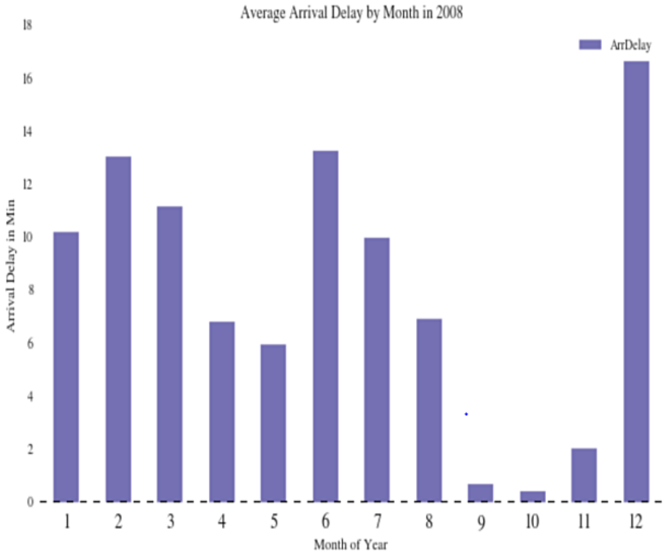


Figure 6.3.4 Average Arrival Distribution

CHAPTER-7

CODING & SOLUTIONING

7.1 Feature 1

HTML CODE:

We have designed the front end of the website. We have included the login page, a page to enter all the details from the user and a page to display the prediction.

Mainpage.html

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML API</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>
<link
href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:30
0' rel='stylesheet' type='text/css'>
```

```
<link rel="stylesheet" href='cssstyle.css'>
</head>
<body>
<div class="logout" style="float:right;">
<a href="login.html"><button>Logout</button></a>
</div>
 <h1 class="h1">Flight Delay Prediction</h1>
 <div class="login">
     <!-- Main Input For Receiving Query to our ML -->
    <form action="Predict.html" method="post">
       Quater of the year <input type="number" name="quater"
placeholder="ex:3" required="required" min='1' max='4' />
       Month in number<input type="number" name="month"
placeholder="ex:12" required="required" min='1' max='12'/>
             Day of the Month<input type="number" name="daym"
placeholder="ex:28" required="required" min='1' max='31'/>
             Day of the week<input type="number" name="dayw"
placeholder="ex:7" required="required" min='1' max='7'/>
       Flight Number<input type="number" name="fnum"
placeholder="ex:2823" required="required" max="9999"/>
             Origin Airport: <select name="airport" class="origin">
      <option value='1'>ATL</option>
      <option value='2'>DWT</option>
      <option value='3'>JFK</option>
      <option value='4'>MSP</option>
      <option value='5'>SEA</option>
```

```
          </select><br><br>
          Destination Airport: <select name="airportd" class="dest">
     <option value='1'>ATL</option>
     <option value='2'>DWT</option>
     <option value='3'>JFK</option>
     <option value='4'>MSP</option>
     <option value='5'>SEA</option>
          </select><br><br>
          Planned Departure Time(format hhmm)<input type="number"
name="dtime" placeholder="ex:1723" required="required" max="9999"/>
          Planned Arrival Time(format hhmm)<input type="number"
name="atime" placeholder="ex:2023" required="required" max="9999"/>
          Estimated Traveling Time(in munites)<input type="number"
name="ttime" placeholder="ex:180" required="required" max="9999"/>
          Distance(in Kms)<input type="number" name="distance"
placeholder="ex:2500" required="required"min='140' max="99999"/>
      <button type="submit" class="btn btn-primary btn-
block">Predict</button>
   </form>
  <br>
  <br>
  {{ prediction_text }}
  <br>
  <br>
  <br>
 </div>
```

```
</body>
</html>
```

login.html

```
<!DOCTYPE html>
<html>
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML API</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>
<link
href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:30
0' rel='stylesheet' type='text/css'>
<link rel="stylesheet" href='Lcssstyle.css'>
</head>
<body>
 <h1 class="h1">LOGIN</h1>
 <div class="login">
    <form action="mainpage.html" method="post">
```

```
        Email: <input type="email" name="email"
placeholder="ex:abc@gmail.com" required="required" autocomplete />
        Password: <input type="password" name="pswd" id="pswd"
placeholder="ex:abc@123" required="required" maxlength=15/>
              {{error}}
               <span id="error" style="color:#F00;"> </span>
        <button type="submit" class="btn btn-primary btn-block"
id="lbtn">login</button>
        <h5>not registered yet </h5><a href="register.html">click here for
register</a>
    </form>
   <br>
   <br>
 </div>
</body>
</html>
```

predict.html

```
<!DOCTYPE html>
<html lang=""en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Flight Delay Prediction</title>
    <!-- <link rel="stylesheet" href="predict.css"-->
    <link rel="stylesheet" type="text/css" href="Predict.css">
    </head>
    <body>
        <h1>Predicated Analysis of your Flight</h1>
        <h2>{{showcase}}</h2>
        <a href="/">Go Back</a>
    </body>
    </html>
```

predict.css

```
*{
    margin:0%;
    padding:0%;
}
body{
    font-family: "Times New Roman", Times, serif;
    background-image:url('flight1.jpg');
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-size: cover;
    text-align: center;
}
h1{
    margin-top: 10px;
    font-size: 55px;
    margin-bottom: 20px;
    text-align: center;
    letter-spacing: 2px;
    font-weight: 300px;
    color: rgb(255, 255,255);
    opacity: .9;
}
h2{
    margin-top: 10px;
    font-size: 120px;
    margin-bottom: 5px;
    text-align: center;
    font-weight: 30px;
    color: rgb(2, 48,94);
    opacity: .9;
}
a{
    font-size: 35px;
}
```

cssstyle.css

```
@import url(https://fonts.googleapis.com/css?family=Open+Sans);
.btn
{
display: inline-block;
*display: inline;
*zoom: 1;
 padding: 4px 10px 4px;
margin-bottom: 0;
```

```
font-size: 13px;
 line-height: 18px;
color: #333333;
text-align: center;
text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75);
 vertical-align: middle;
 background-color: #f5f5f5;
background-image: -moz-linear-gradient(top, #ffffff, #e6e6e6);
background-image: -ms-linear-gradient(top, #ffffff, #e6e6e6);
background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#ffffff),
to(#e6e6e6)); background-image: -webkit-linear-gradient(top, #ffffff,
#e6e6e6);
background-image: -o-linear-gradient(top, #ffffff, #e6e6e6);
 background-image: linear-gradient(top, #ffffff, #e6e6e6);
background-repeat: repeat-x;
filter: progid:dximagetransform.microsoft.gradient(startColorstr=#ffffff,
endColorstr=#e6e6e6, GradientType=0);
 border-color: #e6e6e6 #e6e6e6 #e6e6e6;
border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);
border: 1px solid #e6e6e6;
-webkit-border-radius: 4px; -moz-border-radius: 4px;
border-radius: 4px; -webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255,
0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
 -moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0,
0, 0, 0.05);
```

```
box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0,
0.05);
cursor: pointer; *margin-left: .3em;
}
.btn:hover, .btn:active, .btn.active, .btn.disabled, .btn[disabled]
{
 background-color: #e6e6e6;
}
.btn-large
{
 padding: 9px 14px;
font-size: 15px;
line-height: normal;
-webkit-border-radius: 5px;
-moz-border-radius: 5px;
border-radius: 5px;
}
.btn:hover
{
 color: #333333;
text-decoration: none;
background-color: #e6e6e6;
background-position: 0 -15px;
-webkit-transition: background-position 0.1s linear;
-moz-transition: background-position 0.1s linear;
-ms-transition: background-position 0.1s linear;
```

```
-o-transition: background-position 0.1s linear;
transition: background-position 0.1s linear;
}
.btn-primary, .btn-primary:hover
{
 text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
color: #ffffff;
}
.btn-primary.active
{
color: rgba(255, 255, 255, 0.75);
 }
.btn-primary { background-color: #4a77d4;
background-image: -moz-linear-gradient(top, #6eb6de, #4a77d4);
background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4);
background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#6eb6de),
to(#4a77d4)); background-image: -webkit-linear-gradient(top, #6eb6de,
#4a77d4);
background-image: -o-linear-gradient(top, #6eb6de, #4a77d4);
background-image: linear-gradient(top, #6eb6de, #4a77d4);
background-repeat: repeat-x;
filter: progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de,
endColorstr=#4a77d4, GradientType=0);
border: 1px solid #3762bc;
text-shadow: 1px 1px 1px rgba(0,0,0,0.4);
```

```
box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0,
0.5);
}
.btn-primary:hover, .btn-primary:active, .btn-primary.active, .btn-
primary.disabled, .btn-primary[disabled]
{
 filter: none;
background-color: #4a77d4;
}
.btn-block
{
width: 100%; display:block;
}
*
{
 -webkit-box-sizing:border-box;
-moz-box-sizing:border-box;
-ms-box-sizing:border-box;
-o-box-sizing:border-box;
box-sizing:border-box;
}

html
{
width: 100%;
height:150%;
```

```
}
body
{
	width: 100%;
	height:100%;
	font-family: 'Open Sans', sans-serif;
	background: #092756;
	color: #fff;
	font-size: 16px;
	text-align:center-left;
	letter-spacing:1.2px;
	background: -moz-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%),-moz-linear-gradient(top, rgba(57,173,219,.25) 0%, rgba(42,60,87,.4) 100%), -moz-linear-gradient(-45deg,  #670d10 0%, #092756 100%);
	background: -webkit-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -webkit-linear-gradient(top,  rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -webkit-linear-gradient(-45deg,  #670d10 0%,#092756 100%);
	background: -o-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -o-linear-gradient(top, rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -o-linear-gradient(-45deg, #670d10 0%,#092756 100%);
	background: -ms-radial-gradient(0% 100%, ellipse cover, rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -ms-linear-gradient(top,
rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -ms-linear-gradient(-
45deg,  #670d10 0%,#092756 100%);
        background: -webkit-radial-gradient(0% 100%, ellipse cover,
rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), linear-gradient(to
bottom,  rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), linear-
gradient(135deg,  #670d10 0%,#092756 100%);
        filter: progid:DXImageTransform.Microsoft.gradient(
startColorstr='#3E1D6D', endColorstr='#092756',GradientType=1 );

}
.login {
        position: absolute;
        top: 40%;
        left: 50%;
        margin: -150px 0 0 -150px;
        width:400px;
        height:400px;
}

.h1
 {
font-size: 38px;
color: rgb(238, 87, 87);
text-shadow: 0 0 10px rgba(0,0,0,0.3);
letter-spacing:1px;
text-align:center;
```

```
margin-left: 90px;
 }
body .origin
{
padding-bottom:10px
}
body .dest
{
padding-bottom:10px
}
body .btn
{
padding:20px
}
input
 {
        width: 100%;
        margin-bottom: 10px;
        background: rgba(0,0,0,0.3);
        border: none;
        outline: none;
        padding: 10px;
        font-size: 13px;
        color: #fff;
        text-shadow: 1px 1px 1px rgba(0,0,0,0.3);
        border: 1px solid rgba(0,0,0,0.3);
        border-radius: 4px;
        box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);
        -webkit-transition: box-shadow .5s ease;
        -moz-transition: box-shadow .5s ease;
        -o-transition: box-shadow .5s ease;
        -ms-transition: box-shadow .5s ease;
        transition: box-shadow .5s ease;
}
input:focus { box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px 1px rgba(255,255,255,0.2); }
```

Lcssstyle.css

```
@import url(https://fonts.googleapis.com/css?family=Open+Sans);
.btn
{
 display: inline-block;
*display: inline;
*zoom: 1;
padding: 4px 10px 4px;
margin-bottom: 0;
font-size: 13px;
line-height: 18px;
color: #333333;
text-align: center;
```

```
text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75);
vertical-align: middle;
background-color: #f5f5f5;
background-image: -moz-linear-gradient(top, #ffffff, #e6e6e6);
background-image: -ms-linear-gradient(top, #ffffff, #e6e6e6);
background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#ffffff), to(#e6e6e6));
background-image: -webkit-linear-gradient(top, #ffffff, #e6e6e6);
background-image: -o-linear-gradient(top, #ffffff, #e6e6e6);
background-image: linear-gradient(top, #ffffff, #e6e6e6);
background-repeat: repeat-x;
filter: progid:dximagetransform.microsoft.gradient(startColorstr=#ffffff, endColorstr=#e6e6e6, GradientType=0);
border-color: #e6e6e6 #e6e6e6 #e6e6e6;
border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);
border: 1px solid #e6e6e6;
 -webkit-border-radius: 4px;
 -moz-border-radius: 4px;
border-radius: 4px;
-webkit-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
 -moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0, 0.05);
cursor: pointer;
```

```
*margin-left: .3em;
 }
.btn:hover, .btn:active, .btn.active, .btn.disabled, .btn[disabled]
{
background-color: #e6e6e6;
 }
.btn-large
 {
 padding: 9px 14px;
font-size: 15px;
line-height: normal;
 -webkit-border-radius: 5px;
-moz-border-radius: 5px;
border-radius: 5px;
}
.btn:hover { color: #333333; text-decoration: none; background-color:
#e6e6e6; background-position: 0 -15px;
 -webkit-transition: background-position 0.1s linear; -moz-transition:
background-position 0.1s linear;
 -ms-transition: background-position 0.1s linear;
-o-transition: background-position 0.1s linear;
transition: background-position 0.1s linear;
 }
.btn-primary, .btn-primary:hover { text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
color: #ffffff; }
.btn-primary.active
```

```
{
color: rgba(255, 255, 255, 0.75);
}
.btn-primary
 {
 background-color: #4a77d4;
 background-image: -moz-linear-gradient(top, #6eb6de, #4a77d4);
background-image: -ms-linear-gradient(top, #6eb6de, #4a77d4);
background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#6eb6de),
to(#4a77d4));
 background-image: -webkit-linear-gradient(top, #6eb6de, #4a77d4);
background-image: -o-linear-gradient(top, #6eb6de, #4a77d4);
background-image: linear-gradient(top, #6eb6de, #4a77d4);
 background-repeat: repeat-x; filter:
progid:dximagetransform.microsoft.gradient(startColorstr=#6eb6de,
endColorstr=#4a77d4, GradientType=0);
border: 1px solid #3762bc;
 text-shadow: 1px 1px 1px rgba(0,0,0,0.4);
 box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0, 0, 0,
0.5); }
.btn-primary:hover, .btn-primary:active, .btn-primary.active, .btn-
primary.disabled, .btn-primary[disabled]
 {
 filter: none;
background-color: #4a77d4;
 }
```

```
.btn-block
 {
width: 100%;
display:block;
 }
*
 {
 -webkit-box-sizing:border-box;
 -moz-box-sizing:border-box;
 -ms-box-sizing:border-box;
 -o-box-sizing:border-box;
 box-sizing:border-box;
}
html
{
width: 100%; height:100%;
}
body {
        width: 100%;
        height:100%;
        font-family: 'Open Sans', sans-serif;
        background: #092756;
        color: white;
        font-size: 16px;
        text-align:center-left;
        letter-spacing:1.2px;
        background: -moz-radial-gradient(0% 100%, ellipse cover,
rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%),-moz-linear-gradient(top,
rgba(57,173,219,.25) 0%, rgba(42,60,87,.4) 100%), -moz-linear-gradient(-
45deg,  #670d10 0%, #092756 100%);
        background: -webkit-radial-gradient(0% 100%, ellipse cover,
rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -webkit-linear-
gradient(top,  rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -webkit-
linear-gradient(-45deg,  #670d10 0%,#092756 100%);
        background: -o-radial-gradient(0% 100%, ellipse cover,
rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -o-linear-gradient(top,
rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -o-linear-gradient(-45deg,
#670d10 0%,#092756 100%);
        background: -ms-radial-gradient(0% 100%, ellipse cover,
rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), -ms-linear-gradient(top,
rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), -ms-linear-gradient(-
45deg,  #670d10 0%,#092756 100%);
        background: -webkit-radial-gradient(0% 100%, ellipse cover,
rgba(104,128,138,.4) 10%,rgba(138,114,76,0) 40%), linear-gradient(to
bottom,  rgba(57,173,219,.25) 0%,rgba(42,60,87,.4) 100%), linear-
gradient(135deg,  #670d10 0%,#092756 100%);
        filter: progid:DXImageTransform.Microsoft.gradient(
startColorstr='#3E1D6D', endColorstr='#092756',GradientType=1 );
}
.register {
        position: absolute;
        top: 40%;
```

```
	left: 50%;
	margin: -150px 0 0 -150px;
	width:400px;
	height:400px;
}
.login {
	position: absolute;
	top: 40%;
	left: 50%;
	margin: -150px 0 0 -150px;
	width:400px;
	height:400px;
}
.h1 {font-size: 38px;color: #fff; text-shadow: 0 0 10px rgba(0,0,0,0.3); letter-spacing:1px; text-align:center;margin-left: 90px; }
body .origin{padding-bottom:10px}
body .dest{padding-bottom:10px}
body .btn{padding:20px}
input {
	width: 100%;
	margin-bottom: 10px;
	background: #ffff;
	border: none;
	outline: none;
	padding: 10px;
	font-size: 13px;
```

```
        color: white;
        text-shadow: 1px 1px 1px rgba(0,0,0,0.3);
        border: 1px solid rgba(0,0,0,0.3);
        border-radius: 4px;
        box-shadow: inset 0 -5px 45px rgba(100,100,100,0.2), 0 1px 1px rgba(255,255,255,0.2);
        -webkit-transition: box-shadow .5s ease;
        -moz-transition: box-shadow .5s ease;
        -o-transition: box-shadow .5s ease;
        -ms-transition: box-shadow .5s ease;
        transition: box-shadow .5s ease;
}
input:focus { box-shadow: inset 0 -5px 45px rgba(100,100,100,0.4), 0 1px 1px rgba(255,255,255,0.2); }
```

7.2 Feature 2:

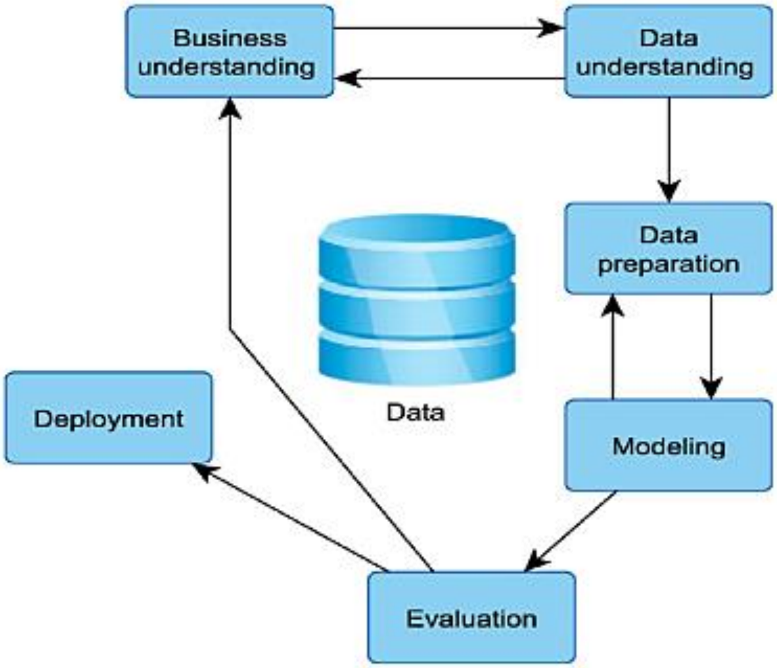
register.html

```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML API</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
```

```
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>
<link
href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:30
0' rel='stylesheet' type='text/css'>
<link rel="stylesheet" href='Lcssstyle.css'>
</head>
<script>
function confirmPass() {
  var pass = document.getElementById("pswd").value
  var confPass = document.getElementById("cpswd").value
  if(pass != confPass) {
    //alert('Wrong confirm password !');
    document.getElementById('error').innerHTML='wrong confirm
password';
          document.getElementById('rbtn').disabled=true;
  }
  else
  {
    document.getElementById('error').innerHTML='';
          document.getElementById('rbtn').disabled=false;
  }
}
//function hello()
```

```
//{
// console.log("Hellow world");
</script>
<body>
 <h1 class="h1">REGISTRATION</h1>
 <div class="register">
    <form action="login.html" method="post">
       Email: <input type="email" name="email"
placeholder="ex:abc@gmail.com" required="required" autocomplete />
       Password: <input type="password" name="pswd" id="pswd"
placeholder="ex:abc@123" required="required" maxlength=15/>
              Confirm Password:<input type="password" name="cpswd"
id="cpswd" placeholder="re-enter same password" required="required"
oninput="confirmPass()" />
              <span id="error" style="color:#F00;"> </span>
       <button type="submit" class="btn btn-primary btn-block"
id="rbtn">Register</button>
    </form>
   <br>
   <br>
 </div>
</body>
     </html>
```

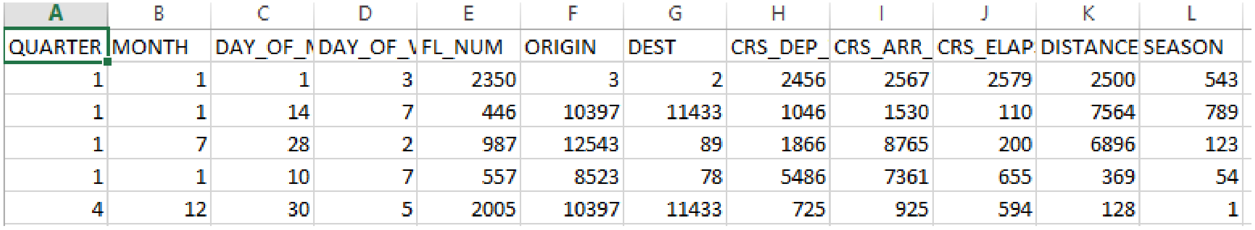
7.3 Database Schema (if Applicable):



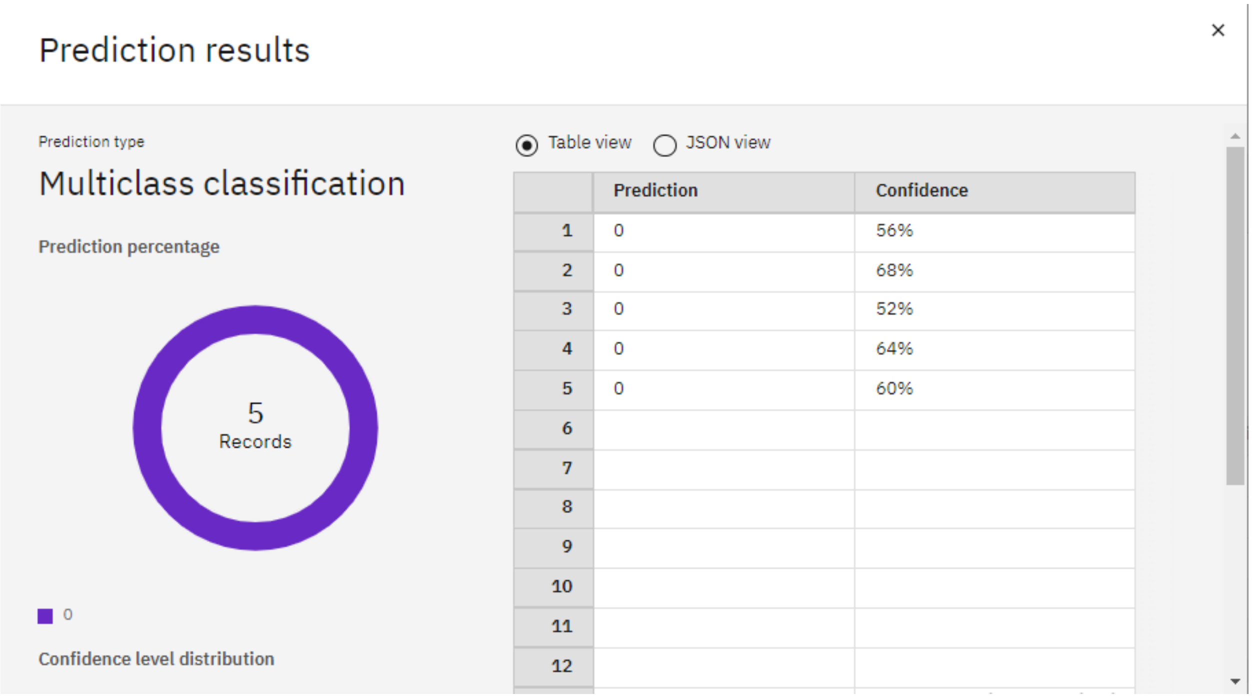
CHAPTER-8

TESTING:

8.1 Test Cases:



A	B	C	D	E	F	G	H	I	J	K	L
QUARTER	MONTH	DAY_OF_I	DAY_OF_V	FL_NUM	ORIGIN	DEST	CRS_DEP_	CRS_ARR_	CRS_ELAP	DISTANCE	SEASON
1	1	1	3	2350	3	2	2456	2567	2579	2500	543
1	1	14	7	446	10397	11433	1046	1530	110	7564	789
1	7	28	2	987	12543	89	1866	8765	200	6896	123
1	1	10	7	557	8523	78	5486	7361	655	369	54
4	12	30	5	2005	10397	11433	725	925	594	128	1



Prediction results ×

Prediction type

Multiclass classification

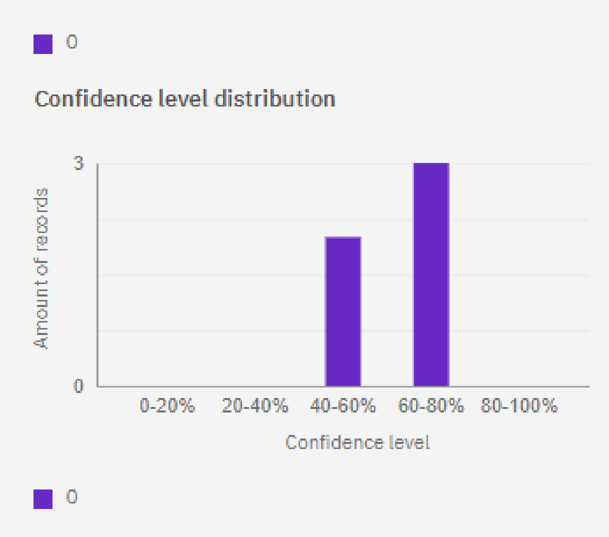
Prediction percentage

■ 0

Confidence level distribution

● Table view ○ JSON view

	Prediction	Confidence
1	0	56%
2	0	68%
3	0	52%
4	0	64%
5	0	60%
6		
7		
8		
9		
10		
11		
12		



8.2 User Acceptance Testing:

8.2.1 Defect Analysis

This report showsthe number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

8.2.2 Test Case Analysis

This report shows the number of test cases that have passed, failed,and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.3.3Test Case Report

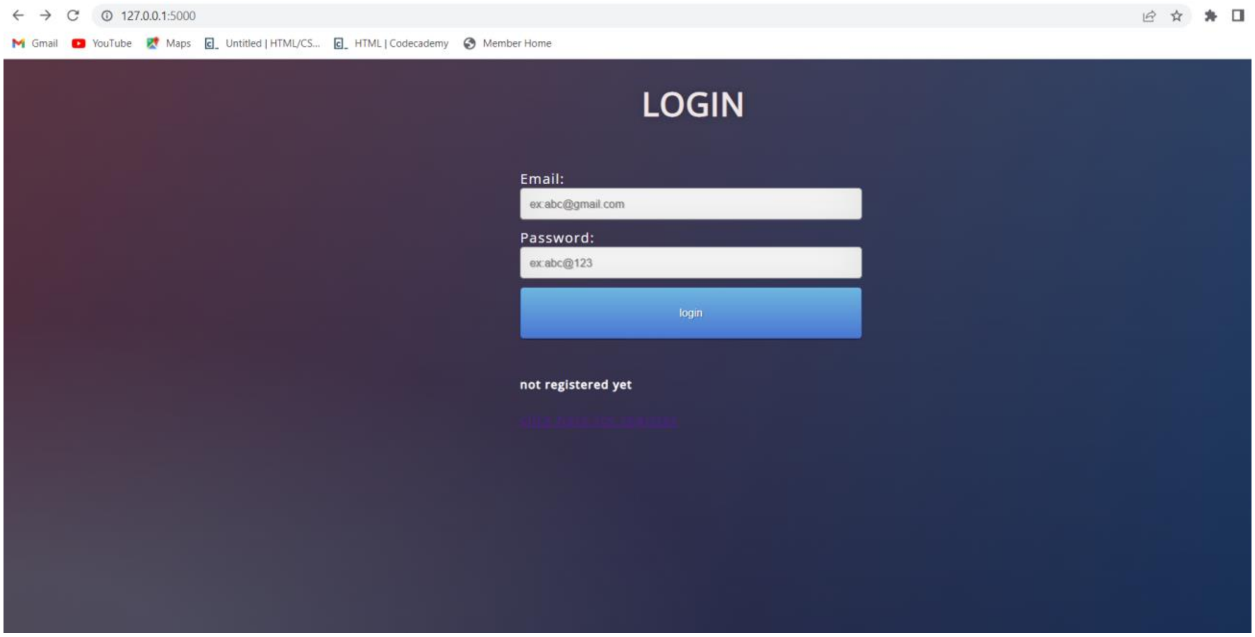
Test case ID	Feature Type	Component	Test Scenario

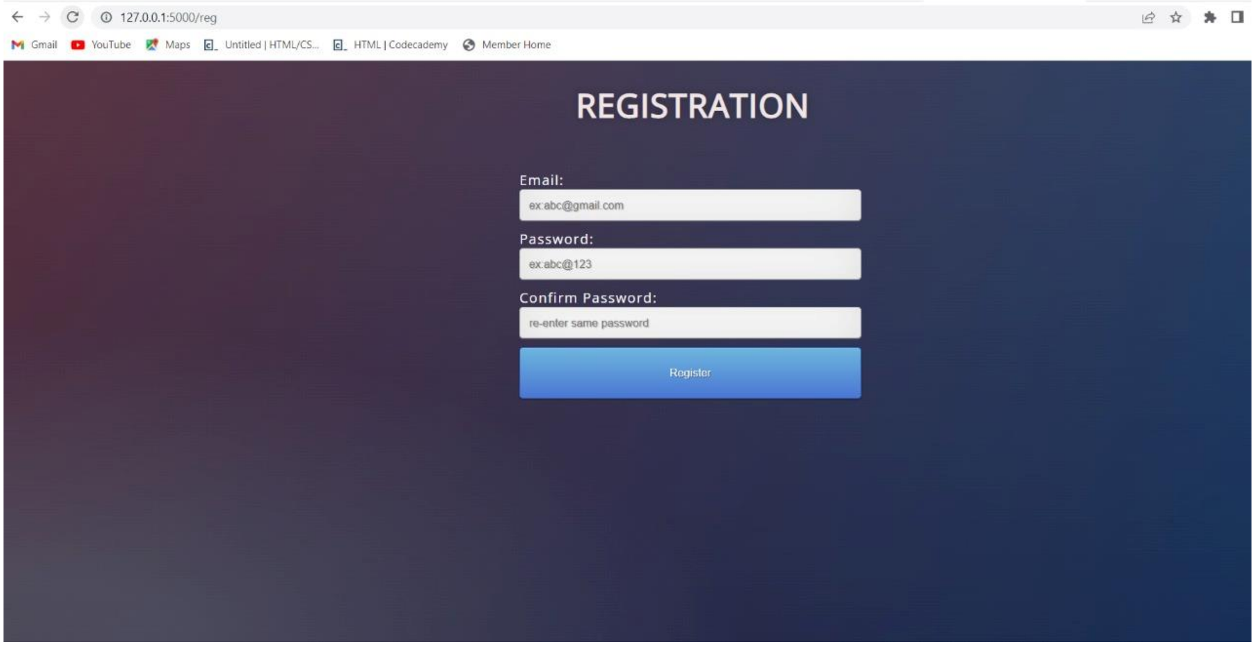
TC_OO1	Functional	Home Page	Verify user able to see the Prediction page
TC_OO2	Functional	Prediction value input page UI	Verify user able to enter input value
TC_OO3	Functional	Prediction value input page UI	Verify user able to enter input value
TC_004	Functional		Predict Rainfall by States
TC_OO5	Functional		Predict Rainfall by States

CHAPTER-9

RESULTS

9.1 Performance Metrics:



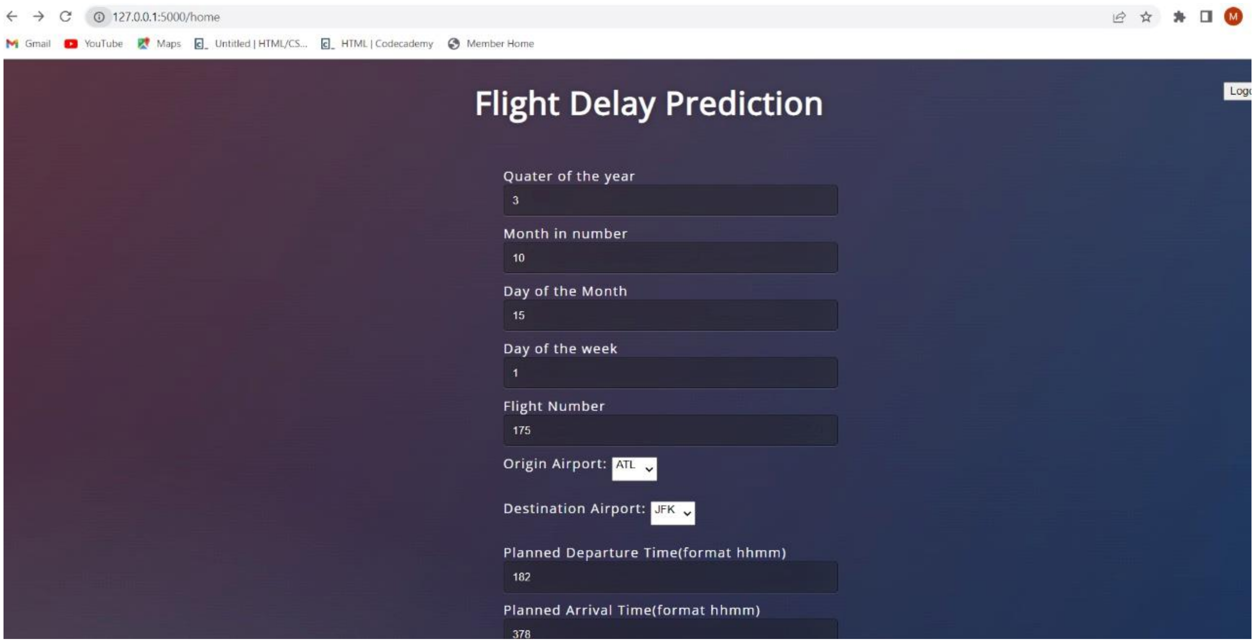


REGISTRATION

Email: ex:abc@gmail.com

Password: ex:abc@123

Confirm Password: re-enter same password



Flight Delay Prediction

Quater of the year 3

Month in number 10

Day of the Month 15

Day of the week 1

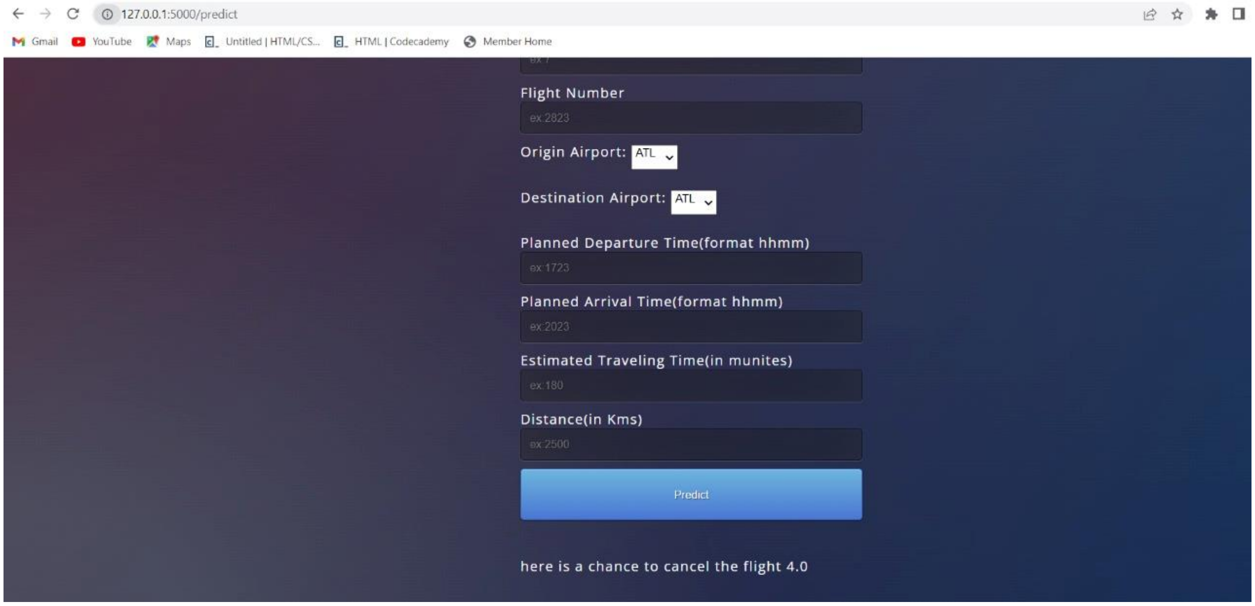
Flight Number 175

Origin Airport: ATL

Destination Airport: JFK

Planned Departure Time(format hhmm) 182

Planned Arrival Time(format hhmm) 378



Flight Number: ex:2823

Origin Airport: ATL

Destination Airport: ATL

Planned Departure Time(format hhmm): ex:1723

Planned Arrival Time(format hhmm): ex:2023

Estimated Traveling Time(in munites): ex:180

Distance(in Kms): ex:2500

here is a chance to cancel the flight 4.0

CHAPTER-10

ADVANTAGES & DISADVANTAGES:

Advantages:

- Predicting flight delays can improve airline operations and passenger satisfaction, which will result in a positive impact on the economy.
- Flight delay is inevitable and it plays an important role in both profits and loss of the airlines.
- An accurate estimation of flight delay is critical for airlines because the results can be applied to increase customer satisfaction and incomes of the airline agencies.
- Multiple users can access the web application without making the server to crash.
- The web application is convenient and adaptable.

Disadvantages:

- The main concern is understanding if websites work well for all kinds of people and in all conditions
- Website deal with very sensitive personal information so app makers need to be able to guarantee privacy for website users.
- The question of who will or should regulate the flight delay prediction and the data it generates needs to be answered.
- Prediction cannot be done without proper information from the passesngers
- Some websites are designed without the involvement of airport officials.

  delays can be induced by different sources and affect airports,airlines, en route airspace or an ensemble of them

CHAPTER-11

CONCLUSION

The Flight Delay Prediction model and the analysis retrieved are useful not only for the passengers but for every decision makers in the aviation industry. Due to flight delay airline can face the financial losses incurred by the industry. It also portray a negative reputation of the airlines and decreases their reliability. The analysis carried out here predicts delays based on the previous available data, but also give statistical description of airlines and delays with respect to time showing the peak hours of delay. This project can be used as a prototype by any aviation authority for their benefit. It can work as an efficient model or a proper prototype to study delay analysis, based on the real dataset provided. This project has encompassed and showed the importance of Regression Analysis in Machine Learning.

CHAPTER-12

FUTURE SCOPE

Further supportive study is required to correlate all the problem, scope and method for getting most accurate result.Although weather conditions are the major reasons for flight delay, other unprecedented events such as major calamities , natural or man-made can cause major delay in flight.

Delays can be induced by different sources and affect airports, airlines, and route airspace or an ensemble of them. For analysis purposes, one may assume a simplified system where only one of these actors or any combination of them is considered .Some work focused on airports to predict delays for all departs considered all airlines.

Airports are also the focus when the objective is to investigate their efficiency based on delays of all carriers . On the other hand, only airlines are considered when comparing the performance of two airlines under the same conditions. An ensemble of airport and route airspace were studied to understand the relationship between congestion and delays. Others considered airports and airlines as well to evaluate capacity problems and airlines decisions. There are many possibilities to ensemble scopes. This becomes important when studying the dynamics of air transportation systems, mainly when targeting root delay.

CHAPTER-13

APPENDIX

13.1 Source Code

```
#import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import rcParams

import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3
def __iter__(self): return 0
# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your
credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='9aSkuMDufRQ-_cGNUoBz7Lt6oBI_rXUiZ_yao-U4oncs',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'flightdelayprediction-donotdelete-pr-ocgt3z2vhznocj'
object_key = 'flightdata2.xlsx'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
df= pd.read_excel(body.read())

df.head()
from datetime import datetime
import datetime as dt
from datetime import datetime

pwd
df.shape
df.info()
df.isnull().any()
df['DEP_DELAY'].fillna(df['DEP_DELAY'].median(),inplace=True)
df.describe()
df.ORIGIN.value_counts()
df.UNIQUE_CARRIER.value_counts()
df.ORIGIN_AIRPORT_ID.value_counts()
df.ORIGIN.unique()
df.ORIGIN_AIRPORT_ID.unique()
df.groupby(by="DAY_OF_WEEK")["DEP_DEL15"].sum()
df.groupby(by="MONTH")["DEP_DEL15"].sum()
df[df["ARR_DELAY"]>=300]
```

```
sm=[6,7,8]
wt=[9,10,11]
sp=[12,1,2,3]
fl=[4,5]
df["SEASON"]=np.where(df["MONTH"].isin(sm),0,np.where(df["MONTH"].isin(wt),1,np.where(df["MONTH"].isin(sp),2,3)))
df["SEASON"].value_counts()
```

Encoding categorial columns into numerical

```
df["CANCELLED"]=np.where(df["CANCELLED"]==1,4,0)
df["DIVERTED"]=np.where(df["DIVERTED"]==1,3,0)
df.CANCELLED.unique(),df.DIVERTED.unique()
df["DELAY_15"]=df["ARR_DEL15"]+df["DEP_DEL15"]
df.DELAY_15.unique()
df["DELAY_15"].fillna(0,inplace=True)
df.DELAY_15.unique()
df["NDELAY"]=df["DELAY_15"]+df["CANCELLED"]+df["DIVERTED"]
df.NDELAY.unique()
len(df["TAIL_NUM"].value_counts()),len(df["FL_NUM"].value_counts())
df.info()
```

Drop unnecessary columns

```
clm=["YEAR","UNIQUE_CARRIER","DEP_TIME","DEP_DELAY","DEP_DEL15","ARR_TIME","ARR_DELAY","ARR_DEL15","CANCELLED","DIVERTED",
    "ACTUAL_ELAPSED_TIME","DELAY_15"]
df1=df.drop(columns=clm)
df1.head()
df1.dropna(how='all',axis=1)
df1.head()
df2=df1.drop(columns=["TAIL_NUM","FL_NUM"])
df2.info()
df2.hist(figsize=(12,12))
df2.NDELAY.value_counts()
```

Spliting Dependent and Independent Variables

```
x=df1.drop(columns=["NDELAY","TAIL_NUM"])
y=df1.NDELAY
```

Encoding Categorical columns to numerical

```
x["ORIGIN"].replace({"ATL":1,'DTW':2,'JFK':3,'MSP':4,'SEA':5},inplace=True)
x["DEST"].replace({"ATL":1,'DTW':2,'JFK':3,'MSP':4,'SEA':5},inplace=True)
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your
credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='9aSkuMDufRQ-_cGNUoBz7Lt6oBI_rXUiZ_yao-U4oncs',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'flightdelayprediction-donotdelete-pr-ocgt3z2vhznocj'
object_key = 'X.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

x1 = pd.read_csv(body)
x1.head()
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your
credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='9aSkuMDufRQ-_cGNUoBz7Lt6oBI_rXUiZ_yao-U4oncs',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'flightdelayprediction-donotdelete-pr-ocgt3z2vhznocj'
object_key = 'Y.csv'
```

```
body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

y2 = pd.read_csv(body)
y2.head()
from sklearn.utils import shuffle
X,Y=shuffle(x1,y2,random_state=72)
X.head()
```

Splitting Dataset as Training and Testing data

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x1,y2,test_size=0.2,random_state=42)
x_train.shape,x_test.shape
```

Model Building

DecisionTree

```
from sklearn.tree import DecisionTreeClassifier
dc=DecisionTreeClassifier()
dc.fit(x_train,y_train)
dc.score(x_test,y_test)
```

RandomForest

```
from sklearn.ensemble import RandomForestClassifier
rf=RandomForestClassifier(n_estimators=50,random_state=42)
rf.fit(x_train,y_train.values.ravel())
rf.score(x_test,y_test)
x_train.shape,x_test.shape
```

Model Building

Decision Tree

```
from sklearn.tree import DecisionTreeClassifier
dc=DecisionTreeClassifier()
dc.fit(x_train,y_train)
dc.score(x_test,y_test)
```

Random Forest

```
from sklearn.ensemble import RandomForestClassifier
rf=RandomForestClassifier(n_estimators=50,random_state=42)
rf.fit(x_train,y_train)
rf.score(x_test,y_test)
pd.DataFrame(rf.predict(x_test)).value_counts()
x_test.iloc[[99,21,22],:]
```

```
y_test.iloc[[8,21,912]]
rf.predict(x_test.iloc[[8,21,912],:])
pd.DataFrame(dc.predict(x_test)).value_counts()
```

Logistic Regression

```
from sklearn.linear_model import LogisticRegression
lr1=LogisticRegression(solver='sag')
lr1.fit(x_train,y_train.values.ravel())
lr1.score(x_test,y_test)
lr1.predict(x_test).sum()
```

SVM

```
from sklearn.svm import SVC
svm=SVC(kernel='sigmoid')
svm.fit(x_train,y_train.values.ravel())
svm.score(x_test,y_test)
pd.DataFrame(svm.predict(x_test)).value_counts()
pd.DataFrame(y_test).value_counts()
```

KNearestNeighborsClassifie

```
from sklearn.neighbors import KNeighborsClassifier
knn=KNeighborsClassifier(n_neighbors=5)
knn.fit(x_train,y_train)
knn.score(x_test,y_test)
pd.DataFrame(knn.predict(x_test)).value_counts()
knn.predict(x_test.iloc[[8,21,912],:])
```

Evaluation of Random Forest

In [103]:

```
from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
pred=rf.predict(x_test)
cm=confusion_matrix(y_test, pred)
plt.figure(figsize=(10,6))
sns.heatmap(cm, annot=True,cmap='winter',linewidths=0.3,
linecolor='black',annot_kws={"size": 20})
TP=cm[0][0]
TN=cm[1][1]
FN=cm[1][0]
FP=cm[0][1]
#print(round(accuracy_score(prediction3,y_test)*100,2))
#print('Testing Accuracy for knn',(TP+TN)/(TP+TN+FN+FP))
```

```
print('Testing Sensitivity for Random Forest',(TP/(TP+FN)))
print('Testing Specificity for Random Forest',(TN/(TN+FP)))
print('Testing Precision for Random Forest',(TP/(TP+FP)))
print('Testing accuracy for Random Forest',accuracy_score(y_test, pred))
print(classification_report(y_test,pred))
```

Evaluation of Decision Tree

In [105]:

```
from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
pred1=rf.predict(x_test)
cm=confusion_matrix(y_test, pred)
plt.figure(figsize=(10,6))
sns.heatmap(cm, annot=True,cmap='winter',linewidths=0.3,
linecolor='black',annot_kws={"size": 20})
TP=cm[0][0]
TN=cm[1][1]
FN=cm[1][0]
FP=cm[0][1]
#print(round(accuracy_score(prediction3,y_test)*100,2))
#print('Testing Accuracy for knn',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Random Forest',(TP/(TP+FN)))
print('Testing Specificity for Random Forest',(TN/(TN+FP)))
print('Testing Precision for Random Forest',(TP/(TP+FP)))
print('Testing accuracy for Random Forest',accuracy_score(y_test, pred1))
print(classification_report(y_test,pred1))
import pickle
pickle.dump(rf,open("rfmodel.pkl",'wb'))
pwd
pred1=dc.predict(x_test)
cm1=confusion_matrix(y_test, pred1)
plt.figure(figsize=(10,6))
sns.heatmap(cm1, annot=True,cmap='winter',linewidths=0.3,
linecolor='black',annot_kws={"size": 20})
TP=cm1[0][0]
TN=cm1[1][1]
FN=cm1[1][0]
FP=cm1[0][1]
#print(round(accuracy_score(prediction3,y_test)*100,2))
print('Testing Accuracy for Decision Tree',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Decision Tree',(TP/(TP+FN)))
print('Testing Specificity for Decision Tree',(TN/(TN+FP)))
print('Testing Precision for Decision Tree',(TP/(TP+FP)))
print('Testing accuracy for Decision Tree',accuracy_score(y_test, pred1))
print(classification_report(y_test,pred))
```

Evaluation of Decision Tree

```
from sklearn.metrics import confusion_matrix,accuracy_score,classification_report
pred=rf.predict(x_test)
cm=confusion_matrix(y_test, pred)
plt.figure(figsize=(10,6))
sns.heatmap(cm, annot=True,cmap='winter',linewidths=0.3,
linecolor='black',annot_kws={"size": 20})
TP=cm[0][0]
TN=cm[1][1]
FN=cm[1][0]
FP=cm[0][1]
#print(round(accuracy_score(prediction3,y_test)*100,2))
#print('Testing Accuracy for knn',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Random Forest',(TP/(TP+FN)))
print('Testing Specificity for Random Forest',(TN/(TN+FP)))
print('Testing Precision for Random Forest',(TP/(TP+FP)))
print('Testing accuracy for Random Forest',accuracy_score(y_test, pred))
print(classification_report(y_test,pred1))
import pickle
pickle.dump(rf,open("rfmodel.pkl",'wb'))
pwd
```

Deployment

```
!pip install -U ibm-watson-machine-learning
from ibm_watson_machine_learning import APIClient
import json
import numpy as np
```

Authenticate and Set Space

```
wml_cred = {
"apikey":"Y7PgzaolxWpOj7WDJSfSSlaKeKNEdrWLhlDda_wEt5Zc",
"url":"https://us-south.ml.cloud.ibm.com"
}
wml_clients=APIClient(wml_cred)
wml_clients.spaces.list()
space_id="494a45e5-4850-459e-b17d-b9e7752c659a"
wml_clients.set.default_space(space_id)
wml_clients.software_specifications.list(500)
```

SAVE AND DEPLOY THE MODEL

```
import sklearn
sklearn.__version__
MODEL_NAME = 'Flight Delay Prediction'
DEPLOYMENT_NAME ='model'
DEMO_MODEL = rf
```

Set Python Version

```
soft_sepc_id=wml_clients.software_specifications.get_id_by_name("runtime-22.1-py3.9")
soft_sepc_id
```

Setup model name

```
model_props={
   wml_clients.repository.ModelMetaNames.NAME:MODEL_NAME,
   wml_clients.repository.ModelMetaNames.TYPE:"scikit-learn_1.0",
   wml_clients.repository.ModelMetaNames.SOFTWARE_SPEC_UID: soft_sepc_id

}
```

Save model

```
model_details = wml_clients.repository.store_model(
 model=rf,
   meta_props=model_props,
   training_data=x_train,
   training_target=y_train
)
model_details
model_uid = wml_clients.repository.get_model_id(model_details)
model_uid
#set meta
deployment_props = {
   wml_clients.deployments.ConfigurationMetaNames.NAME:DEPLOYMENT_NAME,
   wml_clients.deployments.ConfigurationMetaNames.ONLINE: {}
}
deployment=wml_clients.deployments.create(
artifact_uid=model_uid,
meta_props=deployment_props
)
deployment
```

app_ibm.py

```
import numpy as np
import os
from flask import Flask, request, jsonify, render_template,json,redirect,url_for,flash
import pickle
import requests
import sqlite3
```

```
# NOTE: you must manually set API_KEY below using information retrieved from your IBM
Cloud account.
API_KEY = "Y7PgzaolxWpOj7WDJSfSSlaKeKNEdrWLhIDda_wEt5Zc"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
app = Flask(__name__)
app.secret_key="21433253"

conn=sqlite3.connect("database1.db")
conn.execute("CREATE TABLE IF NOT EXISTS login(email TEXT PRIMARY KEY,password
TEXT)")
conn.close()

@app.route('/')
def main():
    return render_template('login.html')

@app.route('/login',methods=['POST','GET'])
def login():
    if request.method=='POST':
        try:
            print("request1")
            fv=[x for x in request.form.values()]
            print(fv)
            print([x for x in request.form.values()])
            print(request.form["email"])
            email=request.form["email"]
            pswd=request.form["pswd"]
            print("request2")
            conn=sqlite3.connect("database1.db")
            cur=conn.cursor()
            print(email,pswd)
            cur.execute("SELECT password FROM login WHERE email=?;",(str(email),))
            print("select")

            result=cur.fetchone()
            cur.execute("SELECT * FROM login")
            print(cur.fetchall())
            print("fetch")
            if result:
                print("login successfully success")
                print(result)
                if result[0]==pswd:
                    flash("login successfully",'success')
                    return redirect('/home')
                else:


   return render_template("login.html", error="please enter correct password")

            else:
                print("register")
                flash("please Register",'danger')

                return redirect('/reg')

        except Exception as e:
            print(e)
            print('danger---------------------------------------------------------------')
            return "hello error"
#    return render_template('login.html')
@app.route('/reg')
def reg():
    return render_template("register.html")

@app.route('/register',methods=['POST','GET'])
def register():
    if request.method=='POST':
        try:
            print("request1")
            fv=[x for x in request.form.values()]
            print(fv)
            print([x for x in request.form.values()])
            print(request.form["email"])
            email=request.form["email"]
            print(request.form["pswd"])
            pswd=request.form["pswd"]
            conn=sqlite3.connect("database1.db")
            print("database")
            cur=conn.cursor()
            print("cursor")
            cur.execute("SELECT * FROM login WHERE email=?;",(str(email),))
            print("fetch")
            result=cur.fetchone()
            if result:
                print("already")
                flash("user already exist,please login",'danger')
                return redirect('/')
            else:
                print("insert")
                cur.execute("INSERT INTO  login(email,password)values(?,?)",(str(email),str(pswd)))
                conn.commit()
                cur.execute("SELECT * FROM login")
                print(cur.fetchall())
                flash("Registered successfully",'success')
                return render_template('login.html')

        except Exception as e:
            print(e)
            #flash(e,'danger')
            return "hello error1"


                #return redirect('/')
   # return render_template('login.html')
@app.route('/home')
def home():
    return render_template('mainpage.html')

@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    sm=[6,7,8]
    wt=[9,10,11]
    sp=[12,1,2,3]
    fl=[4,5]
    farr= [int(x) for x in request.form.values()]
    if farr[1] in sm:


farr.append(0)
    elif farr[1] in wt:
        farr.append(1)
    elif farr[1] in sp:
        farr.append(2)
    else:
        farr.append(3)
    final_features=[int(x) for x in farr]
    print(final_features)
    payload_scoring = {"input_data": [{"fields":
[['QUARTER','MONTH','DAY_OF_MONTH','DAY_OF_WEEK','FL_NUM','ORIGIN','DEST','CRS_DEP_
TIME.1','CRS_ARR_TIME.1','CRS_ELAPSED_TIME','DISTANCE','SEASON']], "values":
[final_features]}]}

    response_scoring = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/595b1b89-5a44-4182-8f21-
9b737ac9ac86/predictions?version=2022-11-14',
json=payload_scoring,headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    pred=response_scoring.json()
    print(pred)
    prediction=pred['predictions'][0]['values'][0][0]
    #prediction = model.predict([final_features])
    print(prediction)

    output =prediction


    if output==0:
        return render_template('mainpage.html', prediction_text='No delay will happen
{}'.format(output))
    elif output==1:
        return render_template('mainpage.html', prediction_text='There is a chance to departure
delay will happen {}'.format(output))
    elif output==2:
        return render_template('mainpage.html', prediction_text='here is a chance to both
departure and arrival delay will happen {}'.format(output))
    elif output==3:
        return render_template('mainpage.html', prediction_text='here is a chance to flight  will
diverted {}'.format(output))
    elif output==4:
        return render_template('mainpage.html', prediction_text='here is a chance to cancel the
flight {}'.format(output))
    else:
        return render_template('mainpage.html', prediction_text='output {}'.format(output))
'''@app.route('/predict_api',methods=['POST'])
def predict_api():

    For direct API calls trought request

    data = request.get_json(force=True)
    prediction = model.predict([np.array(list(data.values()))])
    output = prediction[0]
    return jsonify(output)'''


if __name__ == "__main__":
    os.environ.setdefault('FLASK_ENV', 'development')
    app.run(debug=False)
```

13.2 Github Link

https://github.com/IBM-EPBL/IBM-Project-5601-1658811222