Developing a Flight Delay Prediction Model using Machine Learning

Professional Readiness for Innovation, Employability and Entrepreneurship

Team ID: PNT2022TMID13084

GOKULNATH P K	**(19I217)**
RAJEEVAN V	**(19I245)**
VAIBHAV RAM N	**(19I258)**
VIVITHA L E	**(19I261)**

Dissertation submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

Branch: INFORMATION TECHNOLOGY

of Anna University

NOVEMBER 2022

DEPARTMENT OF INFORMATION TECHNOLOGY

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE – 641 004

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to **Dr.K.Prakasan**, Principal, PSG College of Technology, for giving us the opportunity to do our project with various facilities and infrastructure, without which progress of the task could never have been conceivable.

We extend our heartfelt thanks to **Dr.K.Umamaheswari**, Professor and Head, Department of Information Technology, PSG College of Technology, for her unfailing support throughout this project.

We express our sincere thanks to our Program Coordinator **Dr.D.Karthika Renuka,** Professor, Department of Information Technology, PSG College of Technology, whose constant support and everlasting enthusiasm made it possible to have completed within the time.

We heart fully thank our guide **Dr.S.Sangeetha,**, Assistant Professor, Department of Information Technology, who was always there to help us and played a major role in the completion of the project and we wish to thank her for her enduring guidance and priceless advice throughout this project work.

We also wish to express our sincere thanks to our tutor **Dr.S.Sangeetha,** Assistant Professor, Department of Information Technology, for her guidance that played a vital role in completing my project on time.

Finally, We would like to thank God, Parents, siblings, my faculty members, lab technicians and friends without whom this project work would not have been completed successfully.

SYNOPSIS

Flight Delay Prediction aims to predict the delay in the aircrafts due to the increasing number of travellers in recent times. An aircraft arrival is considered to be delayed if the aircraft is late over 15 minutes between the scheduled time and the arrival time. Flight Delay Prediction takes into consideration various attributes of the delay which includes scheduled time, source and destination of the flight, arrival time of the flight and departure time of the flight and many more attributes to predict the delay in the flight arrivals. These flight delays help the user massively to select the airlines, to select the source station and other economic aspects of the travellers. At the same time, Flight Delay prediction also helps the airlines to focus on the major reasons of the flight delay and minimise delay time on future occasions. The aviation industry is also benefited with the help of the Flight Delay Prediction.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

Flight Delay Prediction aims to predict the delay in the aircrafts due to the increasing number of travellers in recent times. An aircraft arrival is considered to be delayed if the aircraft is late over 15 minutes between the scheduled time and the arrival time. Flight Delay Prediction takes into consideration various attributes of the delay which includes scheduled time, source and destination of the flight, arrival time of the flight and departure time of the flight and many more attributes to predict the delay in the flight arrivals. These flight delays help the user massively to select the airlines, to select the source station and other economic aspects of the travellers. At the same time, Flight Delay prediction also helps the airlines to focus on the major reasons of the flight delay and minimise delay time on future occasions. The aviation industry is also benefited with the help of the Flight Delay Prediction.

1.1 Project Overview

The flight delay prediction mostly depends on the initial steps that are being carried out. Firstly, the dataset collection. The data that is collected for the prediction must be accurate and concise in nature. Any discrepancies in the dataset will cost the accuracy of the flight delay prediction directly. The second step is Data pre-processing. The collected data is improper i.e., those data will have outliers, missing values and the number of attributes may also be huge. At times the data can also be unstructured. In order to solve this issue, the data must be cleaned and pre-processed in a proper manner. The next important issues arise with the data consistency, the flight delay data must be consistent. The time format must be the same across all the dataset. Similarly, the time zone varies from location to location. These inconsistencies must be solved before training the model with the data. The issue also occurs due to abnormalities. For example, the flight delay can be caused due to bad weather or gets cancelled due to any natural calamities. The model cannot predict the flight delays in these abnormalities.

1.2 Purpose

The problem majorly affects the travellers, airline and the aviation industries. The travellers have to look for alternatives in the case of delay of the expected flight arrivals. The airline agency will lose customer trust which is the most important factor. At the same time, Flights are also used to transport goods, the people as well as the organisation who are dependent upon the flight arrival will also be affected due to the flight delay. The aviation industries are also responsible for delivering good products through which the delay of the flights can be reduced. The flight delay prediction may help the aviation industry hugely to protect them from their economic and financial losses. This delay prediction can help the travellers hugely to plan ahead and save their valuable time. The cost associated with the flights can also be majorly decreased when these delays can be predicted correctly. The reputation of the airlines can be majorly dependent on these delay predictions because the delay prediction has a direct hand in determining the customer trust on the airline agency

CHAPTER 2

LITERATURE SURVEY

2. LITERATURE SURVEY

2.1 Existing Problem

Table 2.1 Literature Survey

S.No	**Title**	**Author and year of Publication**	**Proposed Work**	**Limitations**
1.	Airline delay prediction by machine learning algorithms	H. Khaksar et al [1] [2019]	This paper proposes a flight delay prediction model through different methods which includes Bayesian modelling, decision tree, cluster classification, random forest, and hybrid methods. These methods were applied to estimate the occurrences and magnitude of delay in a network.	The accuracy in predicting the flight delay is comparatively low.

2.	Assessing strategic flight schedules at an airport using machine learning-based flight delay and cancellation predictions	Miguel Lambelho et al [2] [2019]	This paper provides a machine learning based approach to assess the strategic flight schedules in terms of potential arrival/departure flight delays and cancellations. This paper also provides an approach that supports an integrated strategic flight schedule assessment, where strategic flight schedules are evaluated with respect to flight delays and cancellations.	This paper has no specific drawbacks.
3.	A Data Mining Approach to Flight Arrival Delay Prediction for American Airlines	Navoneel Chakrabarty [3] [2019]	This paper aims at analysing flight information of US domestic flights operated by American Airlines, covering top 5 busiest airports of US and predicting possible arrival delay of the flight using Data Mining and Machine Learning Approaches.	The data preprocessing should be done better in this proposed work.

4.	A Deep Learning Approach for Flight Delay Prediction through TimeEvolvin g Graphs	Kaiquan Cai et al [4] [2021]	This paper is about the flight delay prediction problem and is investigated from a network perspective (i.e., multi-airport scenario). To model the time-evolving and periodic graph-structured information in the airport network, a flight delay prediction approach based on the graph convolutional neural network (GCN) is developed in this paper .	The quality of model can be improved with efficient data.
5.	Predicting flight delay based on multiple linear regression	Yi Deng et al [5] [2017]	This paper proposes a method to model the arriving flights and a multiple linear regression algorithm to predict delay, comparing with Naive-Bayes and C4.5 approach.	The accuracy and the operational efficiency can be further improved.

6.	Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data	Jingyi Qu et al [6] [2020]	This paper provides two flight delay prediction models using deep convolutional neural networks based on fusion of meteorological data are proposed in this paper. The first model is DCNN (DualChannel Convolutional Neural Network), which refers to the ResNet network structure. The second model is SE-DenseNet (Squeeze and Excitation-Densely Connected Convolutional Network).	Improved Network models can be used to get better results.
7.	Flight Delay Prediction Based on Aviation Big Data and Machine Learning	Guan Gui et al [7] [2020]	This paper explores a broader scope of factors which may potentially influence the flight delay, and compares several machine learning-based models in designed generalised flight delay prediction tasks. To build a dataset for the proposed scheme, automatic dependent surveillance-	The dataset is not sufficient enough to make predicting accuracy higher.

			broadcast (ADS-B) messages are received, preprocessed, and integrated with other information such as weather condition, flight schedule, and airport information.	
8.	Flight delay prediction for commercial air transport: A deep learning approach	Bin Yu et al [8] [2019]	This paper analyses high-dimensional data from Beijing International Airport and presents a practical flight delay prediction model. Following a multifactor approach, a novel deep belief network method is employed to mine the inner patterns of flight delays. Support vector regression is embedded in the developed model to perform a supervised fine-tuning within the presented predictive architecture	The details about air traffic control is not available which is a drawback when comes it to the dataset collection.

9.	Machine Learning Approach for Flight Departure Delay Prediction and Analysis	Ehsan Esmaeilzadeh et al [9] [2020]	This paper employs a support vector machine (SVM) model to explore the nonlinear relationship between flight delay outcomes. Individual flight data were gathered from 20 days in 2018 to investigate causes and patterns of air traffic delay at three major New York City airports	The evaluation metrics used can be improved better.
10.	Development of a predictive model for on-time arrival fight of airliner by discovering correlation between fight and weather data	Noriko Etani [10] [2019]	This paper aims to discover the correlation between flight data and weather data. A predictive model of on-time arrival flight is proposed using flight data and weather data. The feasibility of the predictive model is evaluated by developing a tool of on-time arrival flight prediction.	This paper has no specific drawbacks.

2.2 References

1. H. Khaksar and A. Sheikholeslami, “Airline delay prediction by machine learning algorithms”, Scientia Iranica, Transactions A: Civil Engineering 26 (2019) 2689-2702.

2. Miguel Lambelho, Mihaela Mitici, Simon Pickup, Alan Marsden,”Assessing strategic flight schedules at an airport using machine learning-based flight delay and cancellation predictions”, Journal of Air Transport Management, Volume 82, 2020, 101737, ISSN 0969-6997.

3. Navoneel Chakrabarty,” A Data Mining Approach to Flight Arrival Delay Prediction for American Airlines “, The 9th Annual Information Technology, Electromechanical and Microelectronics Conference (IEMECON 2019).

4. Kaiquan Cai, Yue Li, Yiping Fang, Yanbo Zhu,” A Deep Learning Approach for Flight Delay Prediction through Time-Evolving Graphs”. IEEE Transactions on Intelligent Transportation Systems, IEEE, In press, pp.1-11. ff10.1109/TITS.2021.3103502ff. ffhal-03428046f.

5. Yi Ding,” Predicting flight delay based on multiple linear regression” ,2017 IOP Conf. Ser.:
Earth Environ. Sci. 81 012198

6. Qu, J., Zhao, T., Ye, M. et al. “Flight Delay Prediction Using Deep Convolutional Neural Network Based on Fusion of Meteorological Data.”, Neural Process Lett 52, 1461–1484 (2020).

7. G. Gui, F. Liu, J. Sun, J. Yang, Z. Zhou and D. Zhao, "Flight Delay Prediction Based on Aviation Big Data and Machine Learning.", in IEEE Transactions on

Vehicular Technology, vol. 69, no. 1, pp. 140-150, Jan. 2020, doi: 10.1109/TVT.2019.2954094.

8. Yu, Bin; Guo, Zhen; Asian, Sobhan; Wang, Huaizhu; Chen, Gang (2019),” Flight delay prediction for commercial air transport: A deep learning approach.” Transportation Research Part E: Logistics and Transportation Review.

9. Esmaeilzadeh, Ehsan; Mokhtarimousavi, Seyedmirsajad (2020). “Machine Learning Approach for Flight Departure Delay Prediction and Analysis”. Transportation Research Record: Journal of the Transportation Research Board.

10. Etani, Noriko (2019),” Development of a predictive model for on-time arrival flight of airliner by discovering correlation between flight and weather data.”, Journal of Big Data,2019.

2.3 Problem Statement Definition

Flight Delay Prediction aims to predict the delay in the aircrafts due to increasing number of travellers in the recent times. An aircraft arrival is considered to be delay if the aircraft is late by over 15 minutes between the scheduled time and the arrival time. Flight Delay Prediction takes into consideration various attributes of the delay which includes scheduled time, source and destination of the flight, arrival time of the flight and departure time of the flight and many more attributes to predict the delay in the flight arrivals. These flight delays help the user massively to select the airlines, to select the source station and other economic aspects of the travellers. At the same time, Flight Delay prediction also helps the airlines to focus on the major reasons of the flight delay and minimize delay time on future occasions. Aviation industry are also benefitted with the help of the Flight Delay Prediction.

CHAPTER 3

IDEATION AND PROPOSED SYSTEM

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

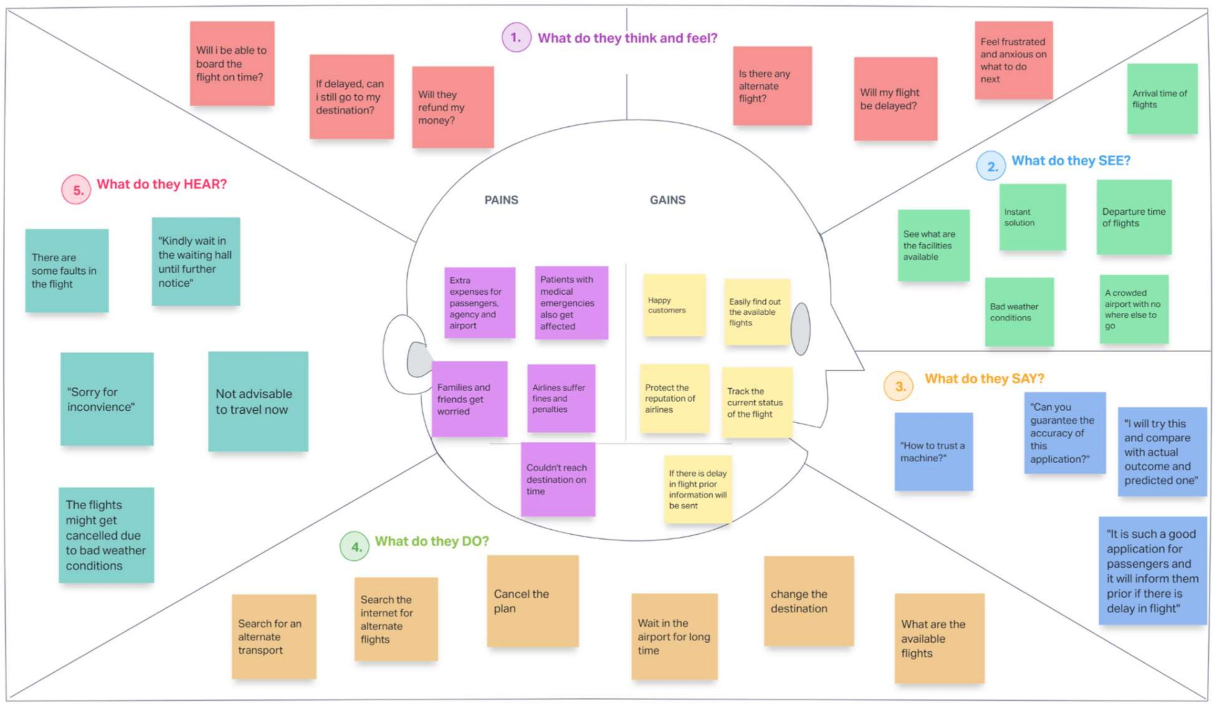


Figure 3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

Brainstorming:

Figure 3.2 Brainstorming

Group Ideas:

Figure 3.3 Group ideas

Prioritise:

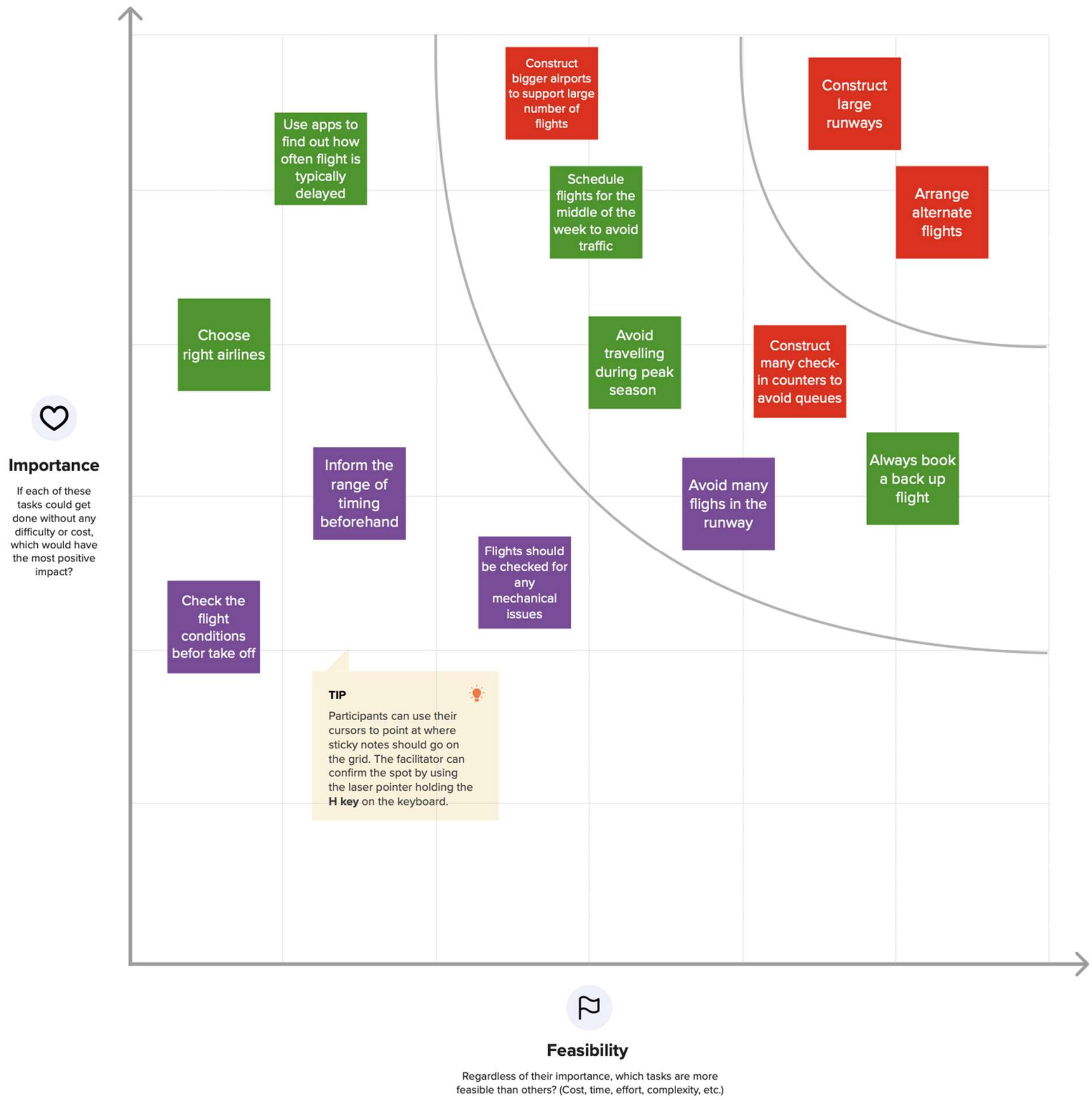


Figure 3.4 Prioritise

3.3 Proposed Solution

The main objective of the model is to predict flight delays accurately in order to optimise flight operations and minimise delays. Using a machine learning model, prediction of flight arrival delays can be done. The input to the algorithm is rows of feature vectors like departure date, departure delay, distance between the two airports, scheduled arrival time etc. Then a decision tree classifier is used to predict if the flight arrival will be delayed or not. Unlike other models here, comparison of decision tree classifiers with logistic regression and a simple neural network for various figures of merit is performed for better efficiency of predict

3.4 Problem Solution fit

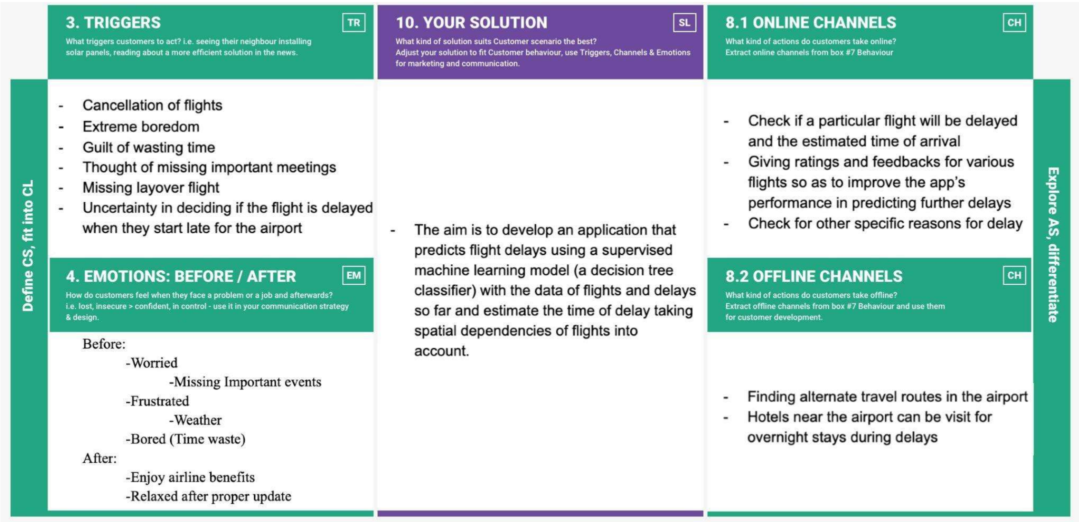


Figure 3.5 Problem Solution Fit

CHAPTER 4

REQUIREMENT ANALYSIS

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirement of proposed system.

4.1 Functional requirements

Table 4.1 Functional Survey

FR No.	**Functional Requirement(Epic)**	**Sub Requirement(Story/Sub-Task)**
FR-1	User Requirements	Use Dataset from google and clean the dataset.
FR-2	User Requirements	Create, test and save the model
FR-3	User Requirements	Display user data entry form to user
FR-4	User Requirements	Receive data from user related to flight arrival time, departure time. These are numeric values
FR-5	User Requirements	Receive data from user. Data to be selected by user.
FR-6	User Requirements	Allow user to click on predict button

FR-7	User Requirements	Display the final result of flight delay or not to the user
FR-8	User Requirements	Deploy model into IBM cloud

Table 4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The application software should be user friendly so many options should be selectable from drop down menu.
NFR-2	Reliability	The software should be tested for same inputs for 20 times and check if output is same.
NFR-3	Performance	The system response should be immediate without any delay
NFR-4	Availability	Software should be always available for success. It should execute graceful degradation.
NFR-5	Scalability	The software can be used for predicting other flight delays just by changing the inputs taken from user.

CHAPTER 5

PROJECT DESIGN

5. PROJECT DESIGN

5.1 Data Flow Diagrams

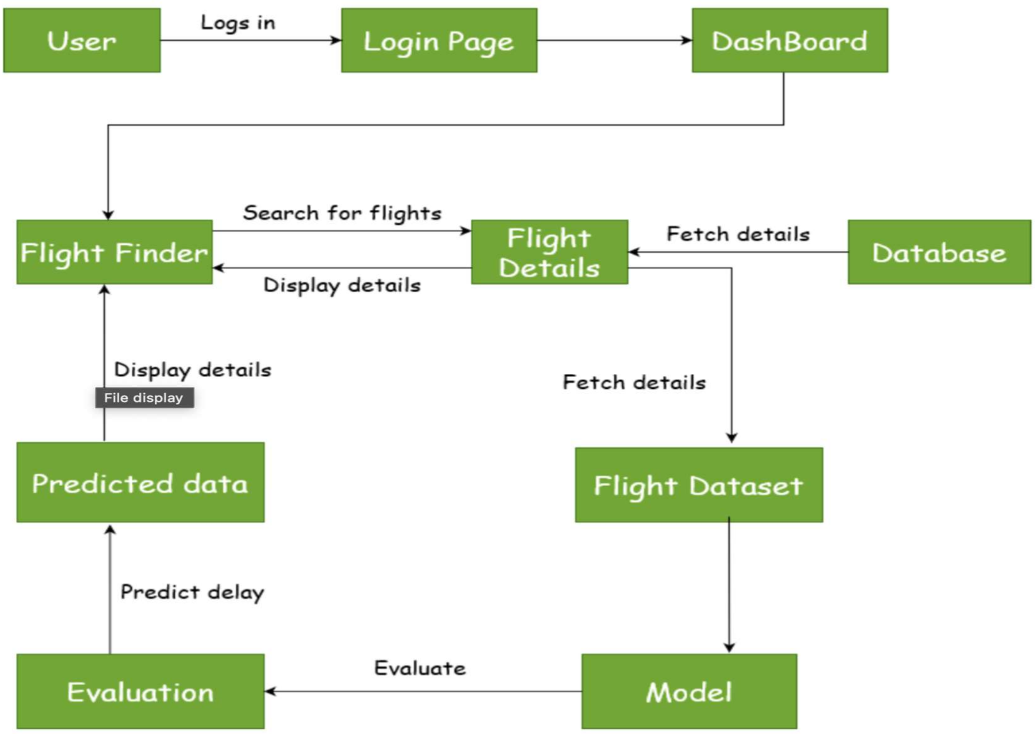


Figure 5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

Solution architecture:

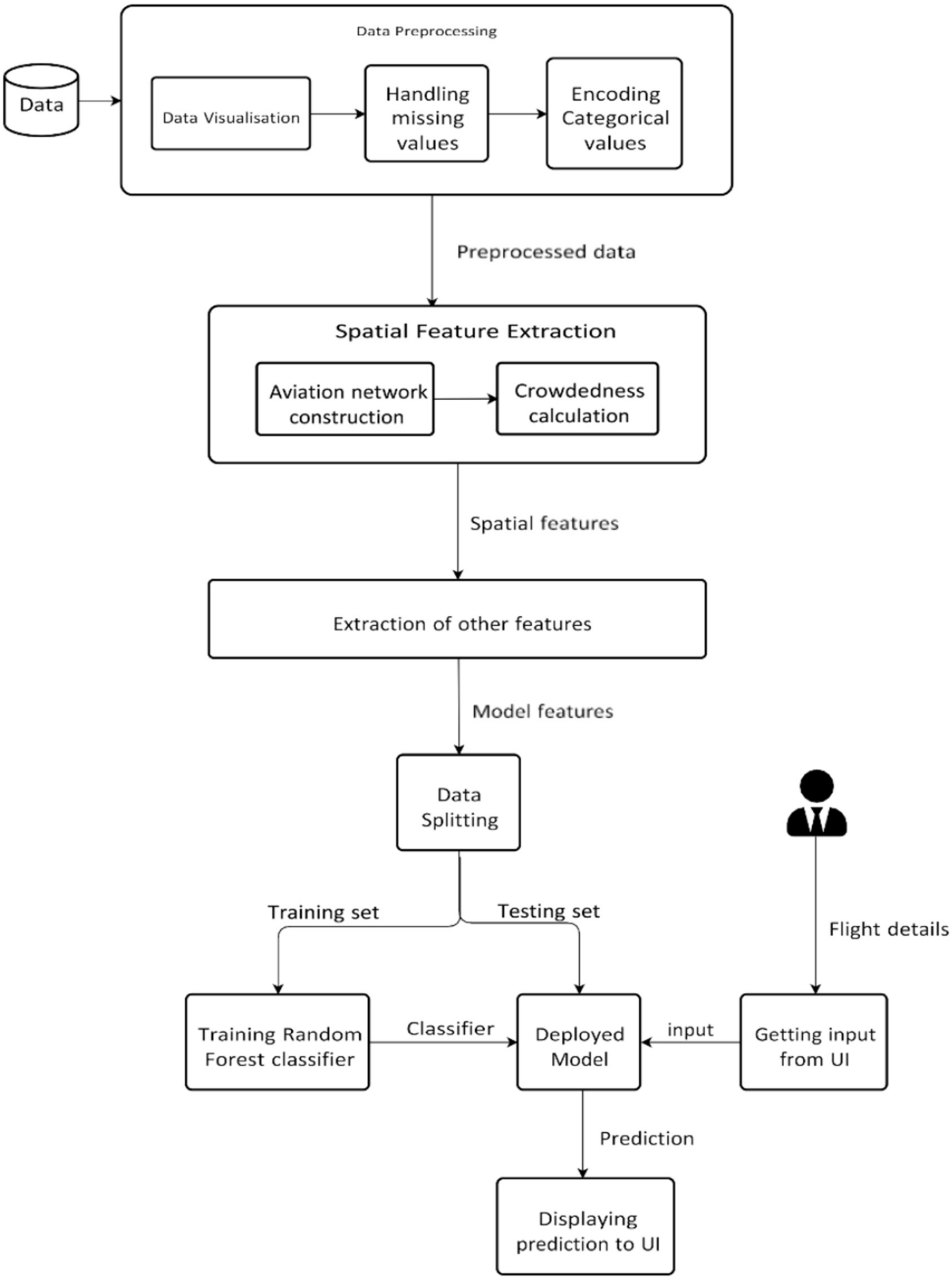


Figure 5.2 Solution Architecture

Technical Architecture:

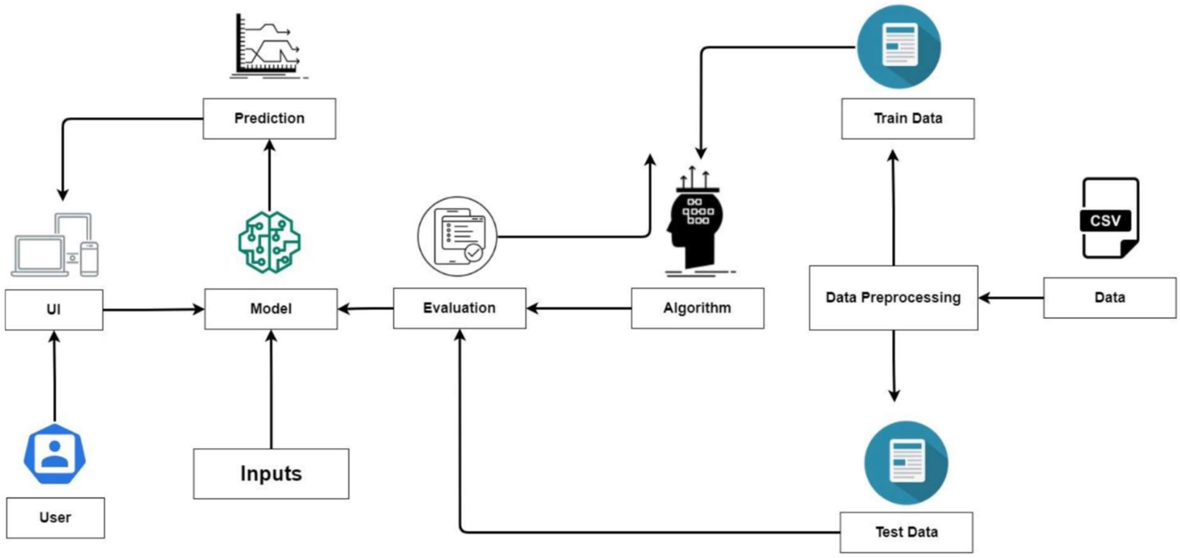


Figure 5.3 Technical Architecture

5.3 User Stories

Table 5.1 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1

		USN-2	As a user, I will receive a confirmation email once I have registered for the application.	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook, Instagram, other social media	I can register & access the dashboard with Facebook/ Instagram Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register and access the dashboard	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can access the dashb oard	High	Sprint-1
	Dashboard	USN-6	As a user, I can	I can access	High	Sprint-1

			navigate through different pages using the dashboard	various pages		
	Search	USN-7	As a user, I can search for flights for different locations	I can receive information on different flights for various locations	High	Sprint-2
	View	USN-8	As a user, I can view the details of flights	I will get the information such as flight no, departure and arrival time, etc.,	High	Sprint-2
	Receive notifications	USN-9	As a user, I will receive notifications about the flight	I will get frequent updates of the flight's location	Low	Sprint-3
	Track	USN-10	As a user, I can track the location of my flight	I can track my flight	Medium	Sprint-3,4
Admin	GPS	USN-11	As an admin, I will need the location of flights	I can track my flight	High	Sprint-3,4
	Analyze	USN-12	As an admin, I will analyse the given dataset	I can analyse the dataset	High	Sprint-2
	Predict	USN-13	As an admin, I will predict the delays	I can predict the flight delays	High	Sprint-2

CHAPTER 6

PROJECT PLANING

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration and Login	USN-1	As a new user, I can register for the application by entering my email and my password.	2	High	Vivitha L E
Sprint-2	Confirmation email	USN-2	As a user, I will receive confirmation email once I have registered for the application	2	Medium	Gokulnath P K
Sprint-1	User login	USN-3	As a user, I can login into the application by entering the registered email-id and password	2	High	Vaibhav Ram N
Sprint-2	Admin Panel	USN-4	As an admin, I can authenticate the registration and login credentials of the passengers	2	High	Rajeevan V
Sprint-3	Arrival and Departure time of flights	USN-5	As a user, I can find all the details of a specific flight with its number or name	2	High	Vivitha L E
Sprint-3		USN-6	As a user, I can find exactly how long the flight will be delayed	2	High	Gokulnath P K
Sprint-4	Helpdesk	USN-7	As a customer care executive, I can provide the contact details of the airlines	1	Medium	Vaibhav Ram N

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4		USN-8	As a passenger, I can find alternative flights to the destination that are available	1	High	Vivitha L E Gokulnath P K Vaibhav Ram N Rajeevan V
Sprint-4	Feedback	USN-9	As a user, I can provide my suggestions and feedback for the improvement of the application	2	Medium	Rajeevan V

Figure 6.1 Sprint planning

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	4	6 Days	27 October 2022	01 November 2022	4	01 November 2022
Sprint-2	4	6 Days	02 November 2022	07 November 2022	4	07 November 2022
Sprint-3	4	6 Days	08 November 2022	13 November 2022	4	13 November 2022
Sprint-4	4	6 Days	14 November 2022	19 November 2022	4	19 November 2022

Velocity:
We have a 24-day sprint duration, and the velocity of the team is 4 (points per sprint). Thus the team's average velocity (AV) per iteration unit (story points per day) is as follows

AV = sprint duration / velocity

= 24/16

= 1.5

Figure 6.2 Sprint Delivery Schedule

Burn down chart:

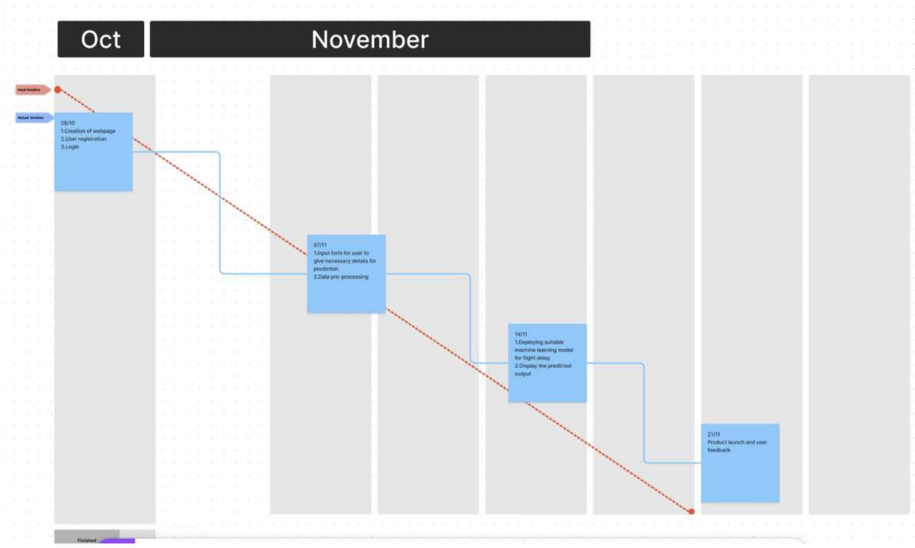


Figure 6.3 Burndown Chart

6.3 Reports from JIRA

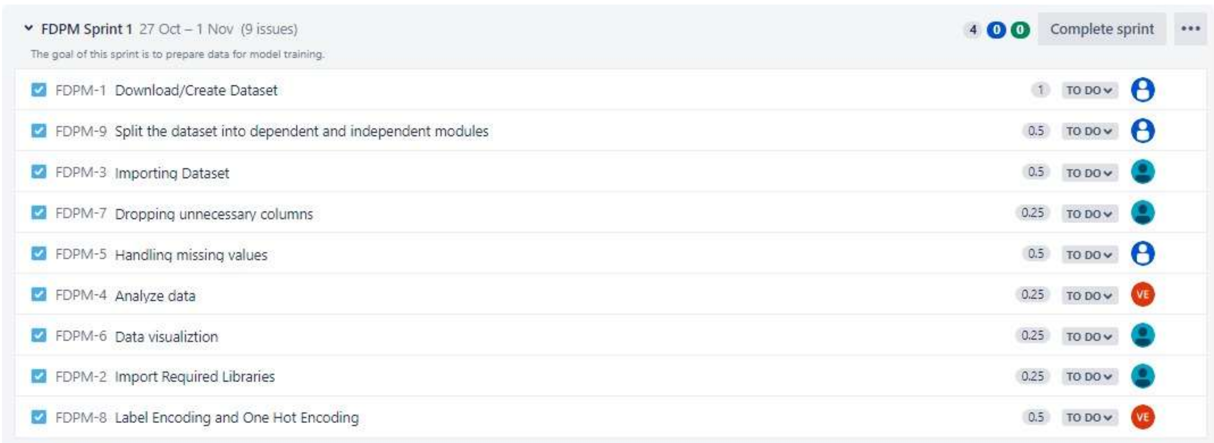


Figure 6.4 Tasks to be performed in Sprint 1

The above figure displays the various tasks to be performed in sprint 1. The goal of the sprint 1 is to prepare data for model training. There are nine issues that need to be addressed in the sprint 1. The story points of each issue is mentioned in the above diagram.

Figure 6.5 Tasks to be performed in Sprint 2

The above figure displays the various tasks to be performed in sprint 2. The goal of the sprint 2 is to train and save the model. There are four issues that need to be addressed in the sprint 2. The story points of each issue is mentioned in the above diagram.

Figure 6.6 Tasks to be performed in Sprint 3

The above figure displays the various tasks to be performed in sprint 3. The goal of the sprint 3 is to build the application and execute the model. There are three issues that need to be addressed in the sprint 3. The story points of each issue are mentioned in the above diagram.

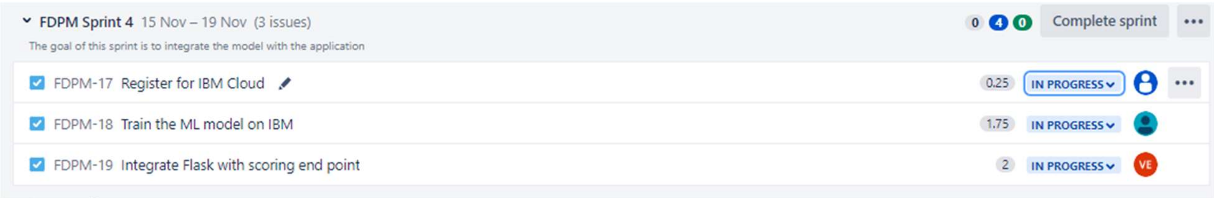


Figure 6.7 Tasks to be performed in Sprint 4

The above figure displays the various tasks to be performed in sprint 4. The goal of the sprint 4 is to integrate the model with the application. There are three issues that need to be addressed in the sprint 3. The story points of each issue are mentioned in the above diagram.

6.3.1 Burnup report

Burnup report maintains the sprint's health by identifying problems such as scope creep or planned path deviation. The burnup reports of each sprint are given below:

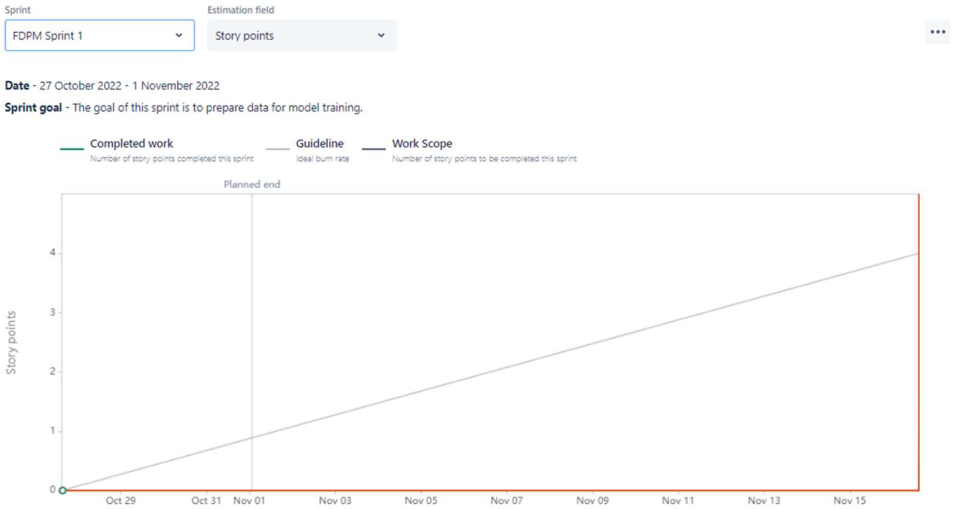


Figure 6.8 Burnup Report of Sprint 1

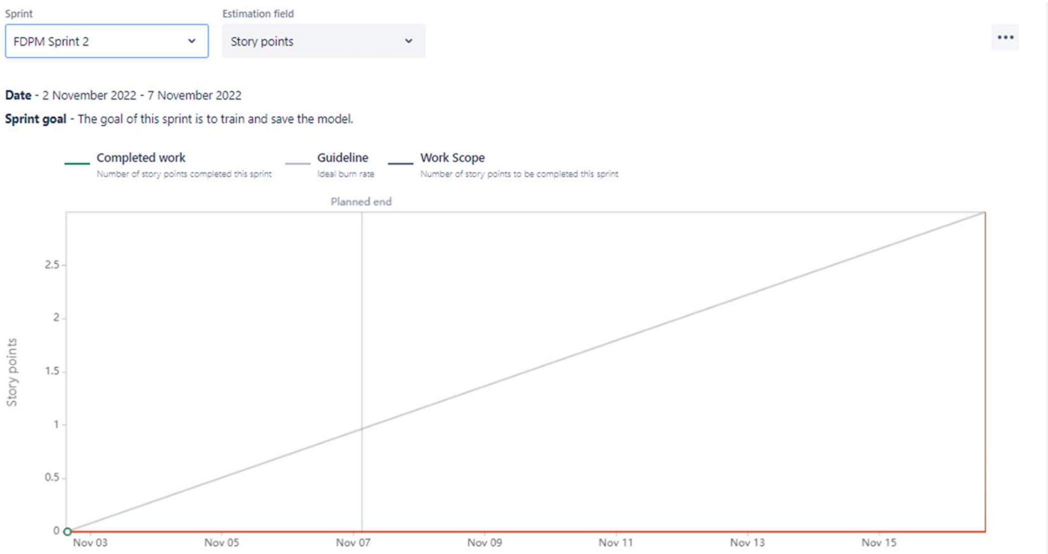


Figure 6.9 Burnup Report of Sprint 2

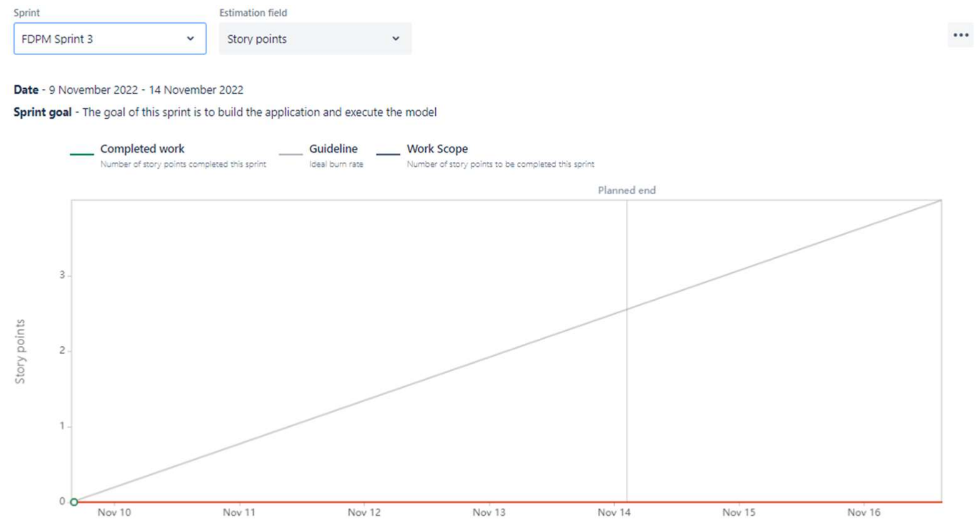


Figure 6.10 Burnup Report of Sprint 3

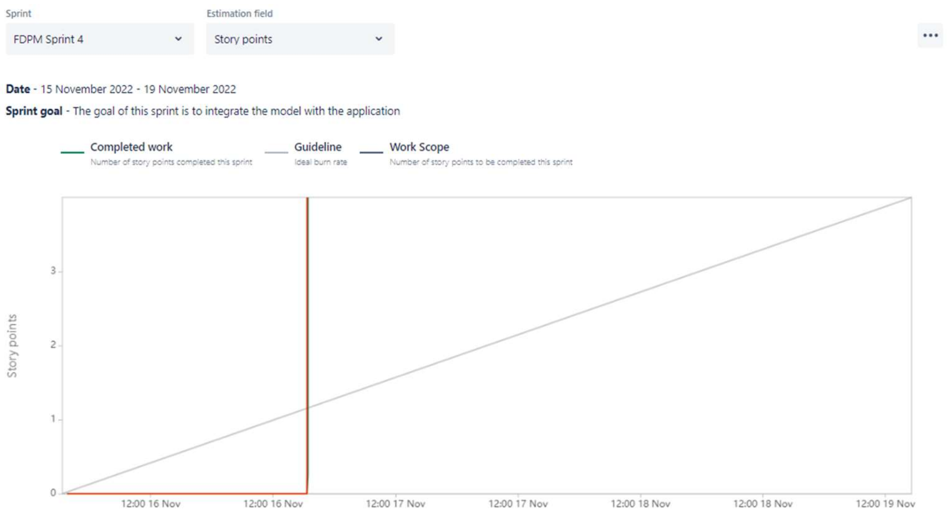
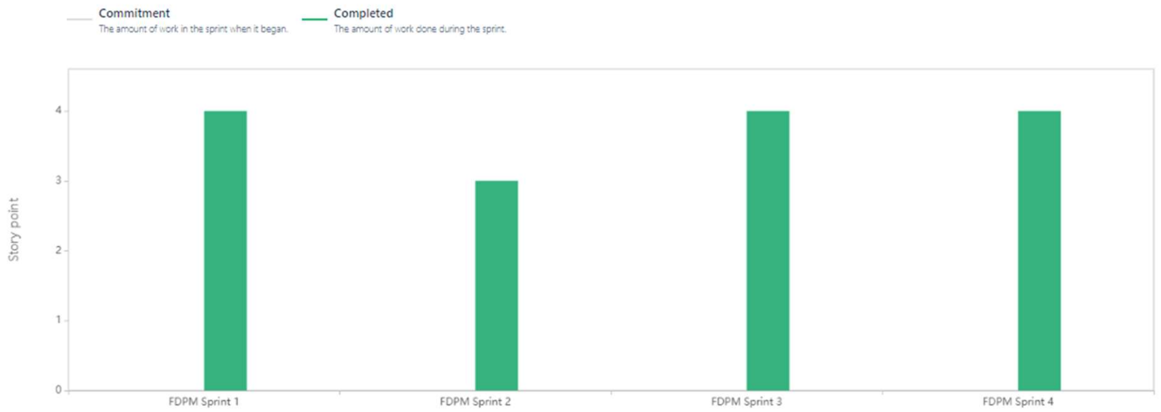


Figure 6.11 Burnup Report of Sprint 4

6.3.2 Velocity report

In the velocity report, the team's velocity is calculated by taking the average of the total completed estimates from their last few sprints. The velocity report of the team is shown below:



Sprint	Commitment	Completed
FDPM Sprint 1	0	4
FDPM Sprint 2	0	3
FDPM Sprint 3	0	4
FDPM Sprint 4	0	4

Figure 6.12 Velocity report

6.3.3 Cumulative flow diagram

Cumulative flow diagram shows the statuses of issues over time. This helps the team identify potential bottlenecks that need to be investigated. The cumulative flow diagram of the team is displayed below

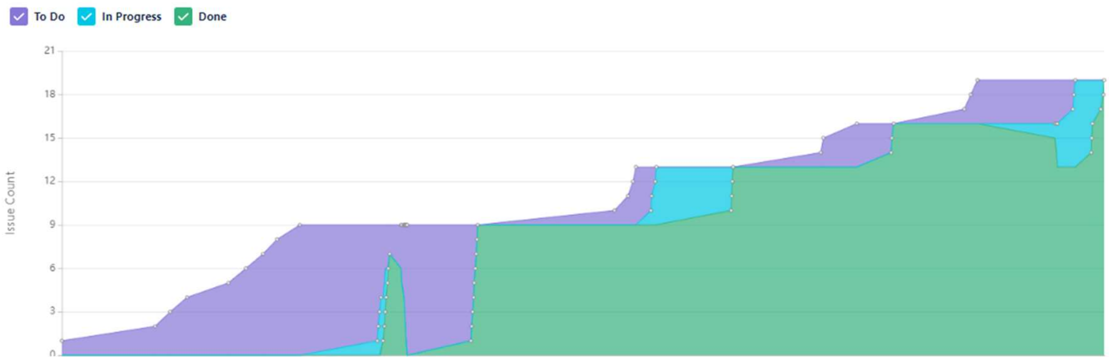


Figure 6.13 Cumulative flow diagram

CHAPTER 7

CODING AND SOLUTIONING

7. CODING & SOLUTIONING

During the Project Development Phase, we have done four Sprints they are Sprint 1, Sprint 2, Sprint 3 and Sprint 4.In Agile product development, a sprint is a set period of time during which specific work has to be completed and made ready for review.

7.1 Sprint 1

During Sprint1 we have planned for Downloading the dataset, import the libraries, Read the dataset, understanding data types and summary of features, handling missing values, Replacing the missing values, Label encoding, Split the dataset into dependent and independent variable, split the dataset into train and test set.

7.2 Sprint 2

During Sprint2 we have planned for Creating HTML files, Build Python code and run the app

Building flask file:

app.py screen shots

```
import numpy as np
import os
from flask import Flask, request, jsonify, render_template
import pickle

app=Flask(__name__)
model = pickle.load(open('rfmodel.pkl', 'rb'))
@app.route("/")
def firstpage():
    return render_template("index.html")
```

```python
@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    summer=[6,7,8]
    Winter=[9,10,11]
    Spring=[12,1,2,3]
    Fall=[4,5]
    Form_Data= [int(x) for x in request.form.values()]
    print(Form_Data[1])
    if Form_Data[1] in summer:
        Form_Data.append(0)
    elif Form_Data[1] in Winter:
        Form_Data.append(1)
    elif Form_Data[1] in Spring:
        Form_Data.append(2)
    else:
        Form_Data.append(3)
    final_features=np.array(Form_Data,dtype='int64')
    print(final_features)
    prediction = model.predict([final_features])
```

```python
    output = round(prediction[0])

    if output==0:
        return render_template('Prediction.html', prediction_text='No delay will happen {}'.format(output))
    elif output==1:
        return render_template('Prediction.html', prediction_text='There is a chance to departure delay will happen {}'.format(output))
    elif output==2:
        return render_template('Prediction.html', prediction_text='here is a chance to both departure and arrival delay will happen {}'.for
    elif output==3:
        return render_template('Prediction.html', prediction_text='here is a chance to flight  will diverted {}'.format(output))
    elif output==4:
        return render_template('Prediction.html', prediction_text='here is a chance to cancel the flight {}'.format(output))
    else:
        return render_template('Prediction.html', prediction_text='output {}'.format(output))
```

Figure 7.1 app.py

Creating HTML files:

Index.html

```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 Data</title>
    <link rel="stylesheet" href="{{ url_for('static',filename='css/index.css')}}">
</head>
<body>
    <h1>Prediction of Flight Delay</h1>
    <form  action="{{ url_for('predict')}}" method="post">
      <div>
        <label for="">Quater of the year</label>
        <input type="number" name="quater" placeholder="ex:3" required="required" min='1' max='4' />

        <label for="">Month:</label>
        <input type="number" name="month" placeholder="ex:12" required="required" min='1' max='12'><br>

        <label for="">Day of Month:</label>
        <input type="number" placeholder="ex:28" required="required" min='1' max='31' name="day" ><br>

        <label for="">Day of Week:</label>
        <input type="number" placeholder="ex:7" required="required" min='1' max='7' name="week"><br>

        <label for="">Enter the Flight Number:</label>
        <input type="number" placeholder="ex:2823" required="required" max="9999" name="flight number" id=""><br>
```

```html
<label for="">Origin:</label>
<select name="origin" id="">
    <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>
<label for="">Destination:</label>
<select name="destination" id="">
  <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>

<label for="">Scheduled Departure Time(format hhmm):</label>
<input type="number"  name="Scheduled dept time" placeholder="ex:1723" required="required" max="9999"><br>

<label for=""> Scheduled Arrival Time(format hhmm):</label>
<input type="number"  placeholder="ex:2023" required="required" max="9999" name="Scheduled arrival time" id=""><br>

<label for="">Actual Departure Time(in minutes):</label>
<input type="number" placeholder="ex:180" required="required" max="9999" name="Actual dept time" id="">
<br>

<label for="">Distance(in Kms):</label>
<input type="number" name="distance" placeholder="ex:2500" required="required"min='140' max="99999"/>
<br>
```

Figure 7.2 index.html

Prediction.html

```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>Prediction Result</title>
    <link rel="stylesheet" href="{{ url_for('static',filename='css/prediction.css')}}">
</head>
<body>
    <h1>Prediction of Flight Delay</h1>
      <div>
        Prediction Result:
        <label for="">{{ prediction_text }}</label>
      </div>
</body>
</html>
```

Figure 7.3 prediction.html

Index.css:

```css
html,body{
    background-image: url("ap.png");
  /* Full height */
  height: 100%;
  /* Center and scale the image nicely */
  background-position: left;
  background-repeat: no-repeat;
  /* background-size: cover; */
}
h1{
    text-align: center;
    font-family: monospace;
}
.button {
    background-color: yellow;
    text-align: center;
  }
  div {
    color: black;
    /* background: rgb(234, 0, 255); */
    padding: 15px;
    position: absolute;
    top: 50%;
    left: 50%;
    -ms-transform: translateX(-50%) translateY(-50%);
    -webkit-transform: translate(-50%,-50%);
    transform: translate(-50%,-50%);
    font-size: 14px;
    display: grid;
line-height: 1.4;
```

Figure 7.4 index.css

7.3 Sprint 3

During Sprint3 we have planned for asking users to enter numerical and selection data and tested for many inputs and checked the correctness of the result.

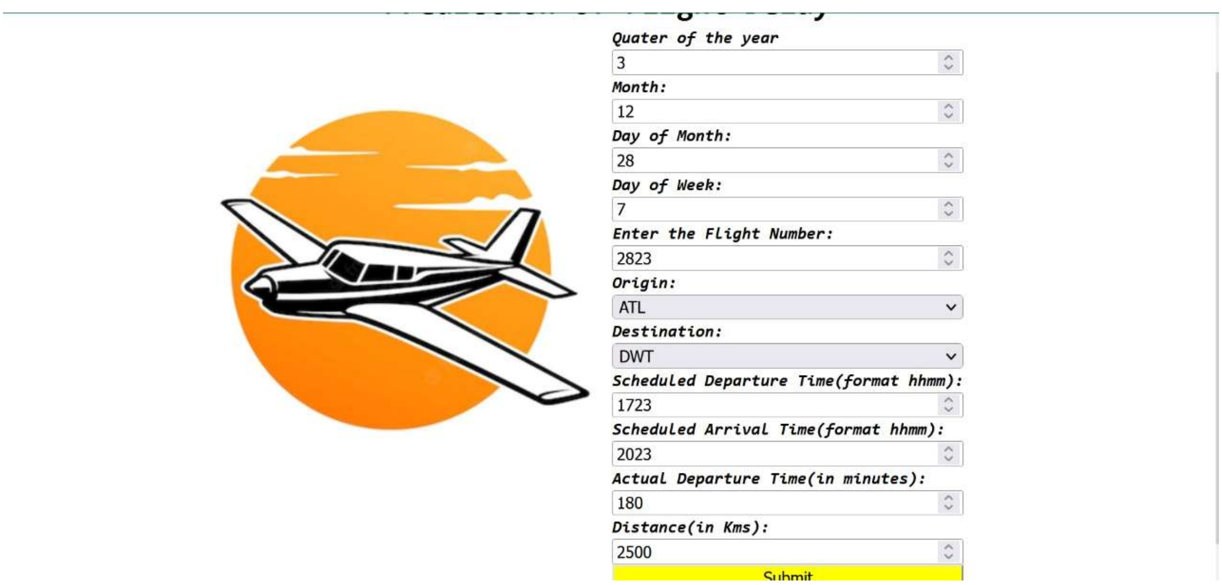


Figure 7.5 Website for flight delay prediction

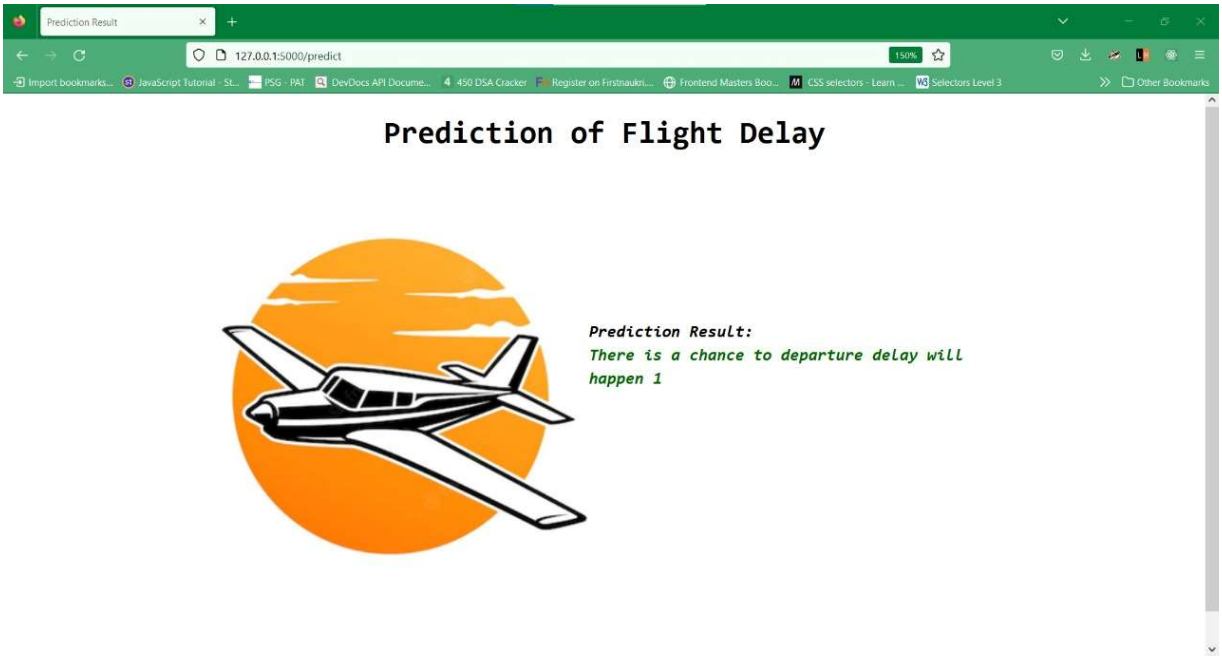


Figure 7.6 The output is predicted whether the flight is delayed or not

7.4 Sprint 4:

During Sprint 4 we have planned for training the model on IBM where we will register for IBM cloud, train the ML model on IBM and integrate flask with scoring end point.

Registered on IBM cloud and activated Watson machine learning, cloud storage and Watson studio then trained the ML model on IBM using API KEY.

In [193]:

```python
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='BGfN6kxTOYC8cVw9eyojsnDinGv0bDBu8u2OJGVPsM18',
    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 = 'flightdelay113-donotdelete-pr-b9qh0sw8dleyxc'
object_key = 'flight-1.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 )

data = pd.read_csv(body)
data.head()
```

In [396]:

```
!pip install -U ibm-watson-machine-learning
```

```
Requirement already satisfied: ibm-watson-machine-learning in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.0.257)
Requirement already satisfied: ibm-cos-sdk==2.11.* in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (2.11.0)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (2022.9.24)
Requirement already satisfied: importlib-metadata in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (4.8.2)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (2.26.0)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (0.8.9)
Requirement already satisfied: pandas<1.5.0,>=0.24.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (1.3.4)
Requirement already satisfied: packaging in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (21.3)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (1.26.7)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (0.3.3)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning) (0.1
0.0)
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning)
(2.11.0)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learni
ng) (2.11.0)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==2.11.*->ib
m-watson-machine-learning) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm-watson-machine-learning) (2021.3)
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm-watson-machine-learning) (1.20.3)
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==
2.11.*->ibm-watson-machine-learning) (1.15.0)
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm-watson-machine-learning) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm-watson-machine-learning) (3.3)
Requirement already satisfied: zipp>=0.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from importlib-metadata->ibm-watson-machine-learning) (3.6.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from packaging->ibm-watson-machine-learning) (3.0.4)
```

Figure 7.6 Training the model in IBM cloud

Authenticate and set space

In [397]:

```python
from ibm_watson_machine_learning import APIClient
```

In [398]:

```python
wml_credentials = {
    "apikey":"UcLivhHTTfB4iebNB-BWzL3XrtMNt9_bDLi3qw0rosnB",
    "url":"https://us-south.ml.cloud.ibm.com"
}
```

In [399]:

```python
wml_client=APIClient(wml_credentials)
wml_client.spaces.list()
```

```
Note: 'limit' is not provided. Only first 50 records will be displayed if the number of records exceed 50
------------------------------------  -------------  ------------------------
ID                                    NAME           CREATED
5a7700b2-ea31-42e3-b588-b2e077f7c0cc  flight_deploy  2022-11-14T11:11:34.221Z
------------------------------------  -------------  ------------------------
```

In [400]:

```python
space_id="5a7700b2-ea31-42e3-b588-b2e077f7c0cc"
```

In [401]:

```python
wml_client.set.default_space(space_id)
```

Out[401]: 'SUCCESS'

Figure 7.7 Authenticate and set space

Save and deploy model

```
In [403]: model_name="demo_model"
          deployment_name="demo_deploy"
          model=dc

In [404]: software_spec_uid=wml_client.software_specifications.get_id_by_name("runtime-22.1-py3.9")

In [405]: model_props={
              wml_client.repository.ModelMetaNames.NAME : model_name,
              wml_client.repository.ModelMetaNames.TYPE : "scikit-learn_1.0",
              wml_client.repository.ModelMetaNames.SOFTWARE_SPEC_UID : software_spec_uid

          }

In [406]: model_details= wml_client.repository.store_model(
              model=model,
              meta_props=model_props,
              training_data=x_train,
              training_target=y_train

          )

In [407]: model_id=wml_client.repository.get_model_id(model_details)
          model_id

Out[407]: '81c72738-41fb-4e79-bbdb-f5442d2cbf71'


In [408]: deployment_props={
              wml_client.deployments.ConfigurationMetaNames.NAME:deployment_name,
              wml_client.deployments.ConfigurationMetaNames.ONLINE: {}
          }

In [409]: deployment=wml_client.deployments.create(
              artifact_uid=model_id,
              meta_props=deployment_props
          )


          #######################################################################################

          Synchronous deployment creation for uid: '81c72738-41fb-4e79-bbdb-f5442d2cbf71' started

          #######################################################################################


          initializing
          Note: online_url is deprecated and will be removed in a future release. Use serving_urls instead.

          ready


          ------------------------------------------------------------------------------------------------
          Successfully finished deployment creation, deployment_uid='7e7da2f7-0679-417e-9f55-9403f2ca3fca'
          ------------------------------------------------------------------------------------------------
```

Figure 7.8 Save and deploy model

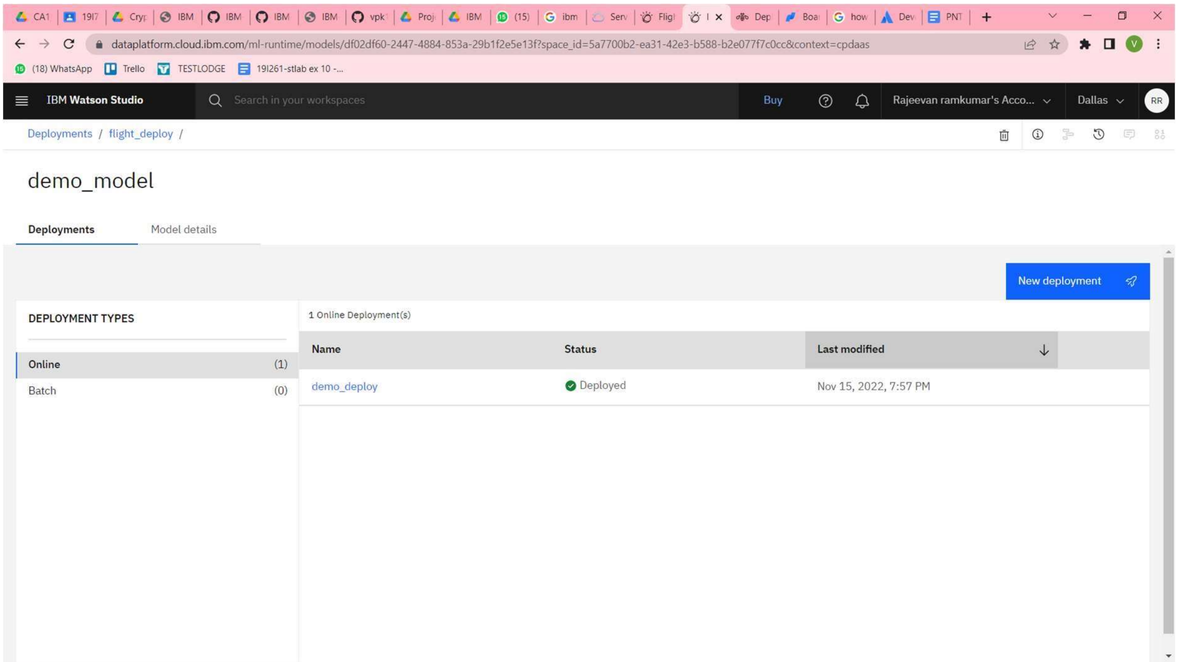


Figure 7.9 The model is successfully deployed in IBM cloud

CHAPTER 8

TESTING

8.TESTING

8.1 Test Cases

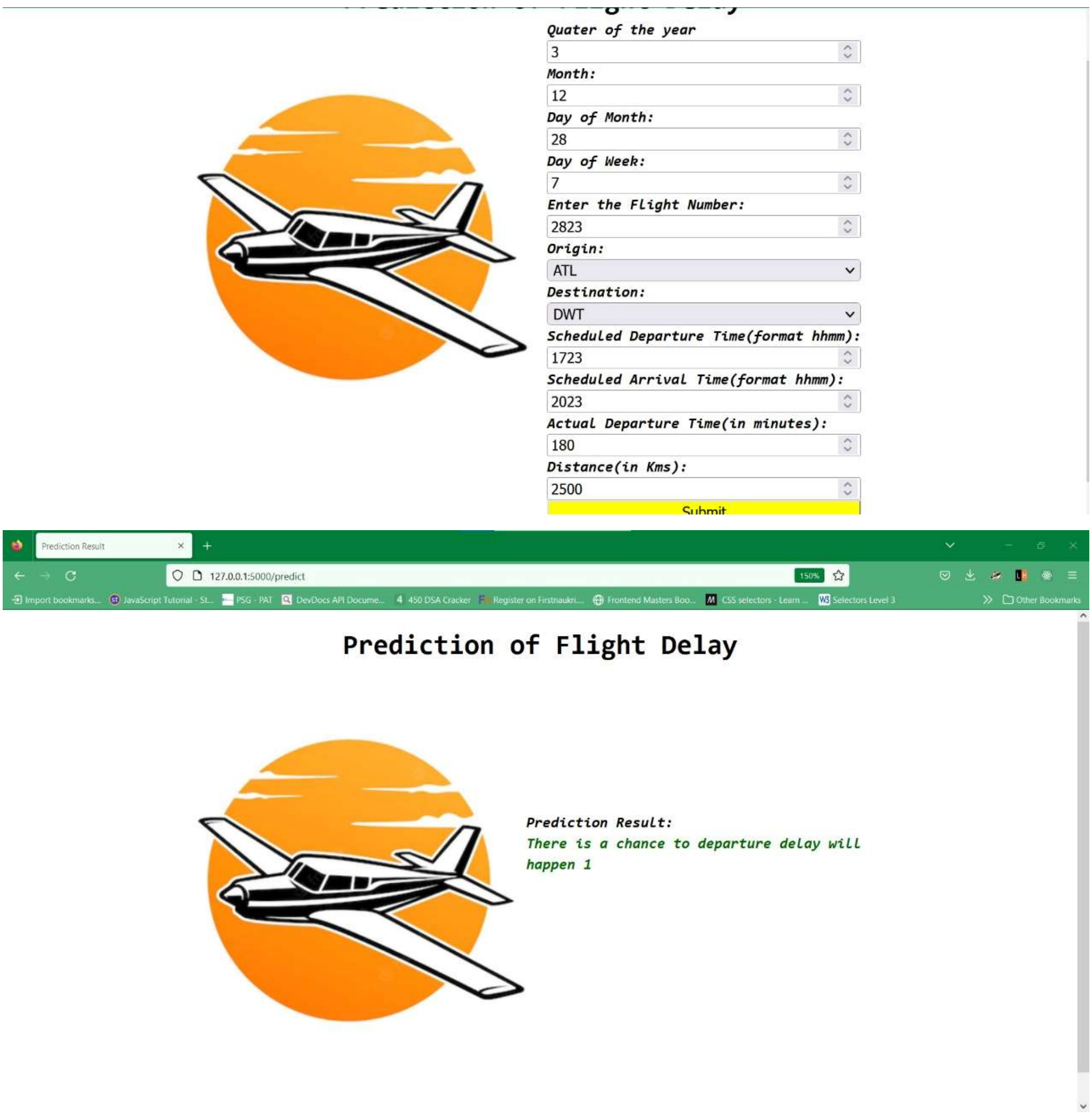


Figure 8.1 Testcase 1

The website predicts if the flight is delayed or not using the given values entered by the user. This helps the user to find the alternative to travel.

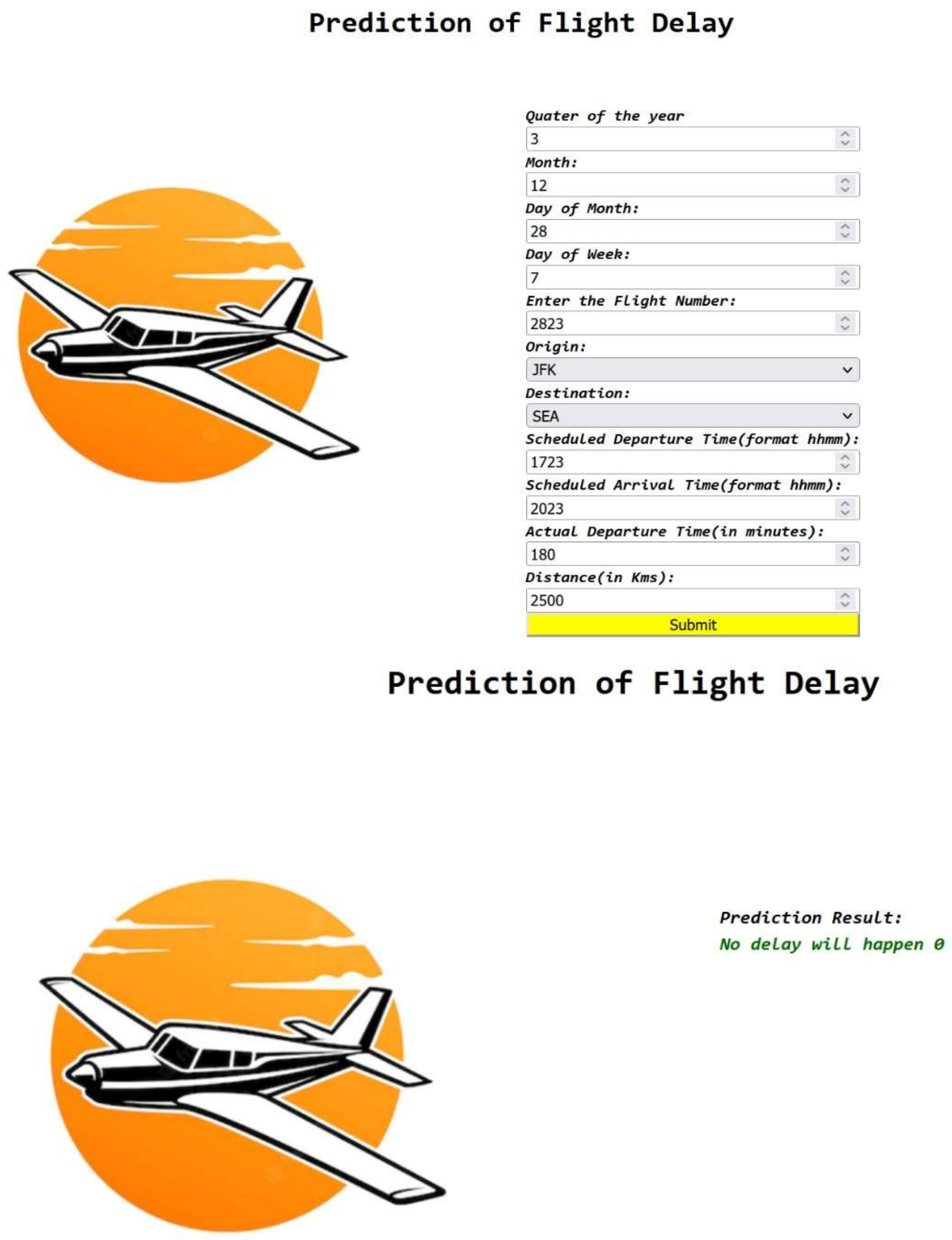


Figure 8.2 Testcase 2

There is no delay in flight so the user can travel in the same flight.

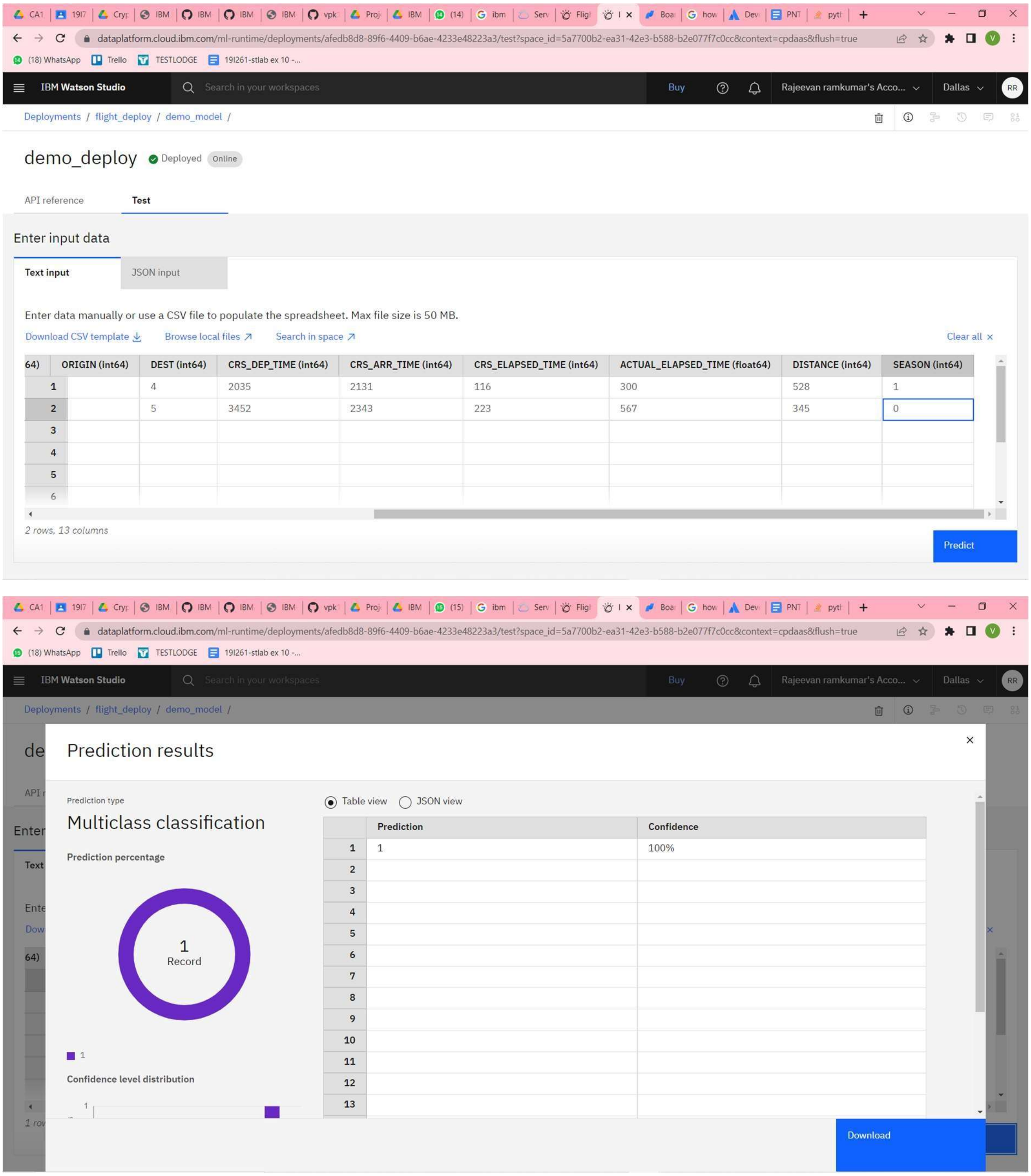


Figure 8.3 The model is tested in the IBM cloud

8.2 User Acceptance Testing

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

Section	Total Cases	Not Tested	Fail	Pass
Home Screen	1	0	0	1
User Input	3	0	0	3
Flight delay testing	2	0	0	2
No Flight delay testing	2	0	0	2
Version Control	2	0	0	2

CHAPTER 9

RESULTS

9. RESULTS

9.1 Performance Metrics

Model: Random Forest Classification performance values

```
[90] accuracy_score(y_test,pred)
0.8095238095238095
```

```
print(classification_report(y_test,pred))#RandomForest
              precision    recall  f1-score   support

         0.0       0.82      0.98      0.89      1797
         1.0       0.22      0.03      0.06       181
         2.0       0.38      0.07      0.12       233
         3.0       1.00      1.00      1.00        15
         4.0       1.00      1.00      1.00        21

    accuracy                           0.81      2247
   macro avg       0.68      0.62      0.61      2247
weighted avg       0.73      0.81      0.75      2247
```

```
conf_mat=confusion_matrix(y_test,pred)
conf_mat
array([[1760,   18,   19,    0,    0],
       [ 166,    6,    9,    0,    0],
       [ 213,    3,   17,    0,    0],
       [   0,    0,    0,   15,    0],
       [   0,    0,    0,    0,   21]])
```

Figure 9.1 Random forest classification metrics

Model: Decision Tree performance values

```
[97] print(classification_report(y_test,pred1))

              precision    recall  f1-score   support

         0.0       0.83      0.80      0.82      1797
         1.0       0.12      0.14      0.13       181
         2.0       0.16      0.18      0.17       233
         3.0       1.00      1.00      1.00        15
         4.0       1.00      1.00      1.00        21

    accuracy                           0.69      2247
   macro avg       0.62      0.62      0.62      2247
weighted avg       0.70      0.69      0.70      2247

[98] accuracy_score(y_test,pred1)

0.6884735202492211

[99] conf_mat=confusion_matrix(y_test,pred1)
     conf_mat

array([[1445,  162,  190,    0,    0],
       [ 137,   25,   19,    0,    0],
       [ 162,   30,   41,    0,    0],
       [   0,    0,    0,   15,    0],
       [   0,    0,    0,    0,   21]])
```

Figure 9.2 Decision tree metrics

Hence, we tested with Decision Tree and Random Forest Classification wherein the accuracy of Random Forest classification is 80% compared with Logistic Regression.

CHAPTER 10

CONCLUSION

10. CONCLUSION

The Flight Delay Prediction model focuses on predicting the delay in the aircrafts due to the increasing number of travellers in recent times. In order to build the flight prediction model, several steps are followed .The database for the flight delay prediction model is taken and analysed in a proper fashion. The required libraries needed for implementing the model are imported. The importance of each attribute is then calculated. The unwanted attributes of the databases are identified and are removed. The missing values in the remaining dataset are handled properly. One hot encoding is performed and then finally the dataset is splitted for training and testing purposes. This is how the dataset is pre-processed. Then, the Random Forest Classifier model is selected. The random forest classifier model is trained on the pre-processed dataset and is evaluated using different metrics. Once the model is evaluated, the model is saved using pickle for further predictions. The accuracy of the Random Forest classifier is around 80%. The appropriate pages of the application are created using HTML and Python.

The saved model is executed and tested with appropriate test cases. Finally, the model is made to train in the IBM cloud. Flask framework (Python) is used to integrate the model and the application. Each user can view the flight delay using this application. The end user will be asked to enter certain values .The end user might be able to view the delay of the flights after giving the appropriate inputs.

CHAPTER 11

FUTURE SCOPE

11. FUTURE SCOPE

The proposed flight delay prediction model has an accuracy of around 80%. The model performs relatively well. Yet, the accuracy of the model can be improved by using advanced machine learning algorithms. The application in the proposed system can be improved by using a user authentication module.

CHAPTER 12

APPENDIX

12. APPENDIX

Source Code:

App.py

```
import
NumPy
as np
import os
from flask import Flask, request, jsonify, render_template
import pickle

app=Flask(__name__)
model = pickle.load(open('rfmodel.pkl', 'rb'))
@app.route("/")
def firstpage():
    return render_template("index.html")

@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    summer=[6,7,8]
    Winter=[9,10,11]
    Spring=[12,1,2,3]
    Fall=[4,5]
    Form_Data= [int(x) for x in request.form.values()]
    print(Form_Data[1])
    if Form_Data[1] in summer:
        Form_Data.append(0)
    elif Form_Data[1] in Winter:
        Form_Data.append(1)
    elif Form_Data[1] in Spring:
        Form_Data.append(2)
    else:
```

```
            Form_Data.append(3)
        final_features=np.array(Form_Data,dtype='int64')
        print(final_features)
        prediction = model.predict([final_features])


        output = round(prediction[0])

        if output==0:
            return render_template('Prediction.html', prediction_text='No delay will
    happen {}'.format(output))
        elif output==1:
            return render_template('Prediction.html', prediction_text='There is a
    chance to departure delay will happen {}'.format(output))
        elif output==2:
            return render_template('Prediction.html', prediction_text='here is a
    chance to both departure and arrival delay will happen {}'.format(output))
        elif output==3:
            return render_template('Prediction.html', prediction_text='here is a
    chance to flight  will diverted {}'.format(output))
        elif output==4:
            return render_template('Prediction.html', prediction_text='here is a
    chance to cancel the flight {}'.format(output))
        else:
            return render_template('Prediction.html', prediction_text='output
    {}'.format(output))



    if __name__=='__main__':
        app.run(debug=True)
```

Prediction.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>Prediction Result</title>
                <link rel="stylesheet" href="{{
            url_for('static',filename='css/prediction.css')}}">
```

```
</head>
<body>
    <h1>Prediction of Flight Delay</h1>
      <div>
        Prediction Result:
        <label for="">{{ prediction_text }}</label>
      </div>
</body>
</html>
```

Index.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 Data</title>
    <link rel="stylesheet" href="{{
url_for('static',filename='css/index.css')}}">
</head>
<body>
    <h1>Prediction of Flight Delay</h1>
    <form  action="{{ url_for('predict')}}" method="post">
      <div>
        <label for="">Quater of the year</label>
        <input type="number" name="quater" placeholder="ex:3"
required="required" min='1' max='4' />

        <label for="">Month:</label>
        <input type="number" name="month" placeholder="ex:12"
required="required" min='1' max='12'><br>

        <label for="">Day of Month:</label>
        <input type="number" placeholder="ex:28" required="required" min='1'
max='31' name="day" ><br>

        <label for="">Day of Week:</label>
        <input type="number" placeholder="ex:7" required="required" min='1'
max='7' name="week"><br>

        <label for="">Enter the Flight Number:</label>
```

```
        <input type="number" placeholder="ex:2823" required="required"
max="9999" name="flight number" id=""><br>

        <label for="">Origin:</label>
        <select name="origin" id="">
            <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>
        <label for="">Destination:</label>
        <select name="destination" id="">
          <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>

        <label for="">Scheduled Departure Time(format hhmm):</label>
        <input type="number"  name="Scheduled dept time" placeholder="ex:1723"
required="required" max="9999"><br>

        <label for=""> Scheduled Arrival Time(format hhmm):</label>
        <input type="number"  placeholder="ex:2023" required="required"
max="9999" name="Scheduled arrival time" id=""><br>

        <label for="">Actual Departure Time(in minutes):</label>
        <input type="number" placeholder="ex:180" required="required"
max="9999" name="Actual dept time" id="">
        <br>

        <label for="">Distance(in Kms):</label>
        <input type="number" name="distance" placeholder="ex:2500"
required="required"min='140' max="99999"/>
        <br>
        <button class="button">Submit</button>
      </div>
    </form>
</body>
</html>
```

Index.css

```
html,body{
                 background-image: url("ap.png");

               /* Full height */
               height: 100%;

               /* Center and scale the image nicely */
               background-position: left;
               background-repeat: no-repeat;
               /* background-size: cover; */
             }
             h1{
                 text-align: center;
                 font-family: monospace;
             }

             .button {
                 background-color: yellow; /* Green */
                 /* border: none; */
                 /* color: white; */
                 /* padding: 15px 32px; */
                 text-align: center;

                 /* text-decoration: none; */
                 /* display: inline-block; */
                 /* font-size: 16px; */
               }

               div {
                 color: black;
                 /* background: rgb(234, 0, 255); */
                 padding: 15px;
                 position: absolute;
                 top: 50%;
                 left: 50%;
                 -ms-transform: translateX(-50%) translateY(-50%);
                 -webkit-transform: translate(-50%,-50%);
                 transform: translate(-50%,-50%);
                 font-size: 14px;
                 display: grid;
             line-height: 1.4;
             /* letter-spacing: 0.149em; */
             font-weight: 800;
```

```
font-style: italic;
margin-left: 150px;
margin-top: 50px;
font-family: monospace;
/* margin-bottom: 140px; */
border-bottom: 9.6px;
  }

input{
    position: relative;
}
```

Prediction.css

```
html,body{
    background-image: url("ap.png");

  /* Full height */
  height: 100%;

  /* Center and scale the image nicely */
  background-position: left;
  background-repeat: no-repeat;
  /* background-size: cover; */
}
h1{
    text-align: center;
    font-family: monospace;
    /* color: tomato; */
}


  div {
    color: black;
    /* background: rgb(234, 0, 255); */
    padding: 15px;
    position: absolute;
    top: 50%;
    left: 50%;
    -ms-transform: translateX(-50%) translateY(-50%);
    -webkit-transform: translate(-50%,-50%);
    transform: translate(-50%,-50%);
    font-size: 14px;
```

```
    display: grid;
line-height: 1.4;
/* letter-spacing: 0.149em; */
font-weight: 800;
font-style: italic;
margin-left: 150px;
margin-top: 0px;
font-family: monospace;
top: 220.067px;
/* margin-bottom: 140px; */
border-bottom: 9.6px;
  }

label{
  color: darkgreen;
}
```

GitHub & Project Demo Link

GitHub link: https://github.com/IBM-EPBL/IBM-Project-12025-1659366516

Demolink:

https://drive.google.com/drive/folders/1fyyQ3wGV70PnzWIbjjKWSAIbBr5LEhVF?usp=sharing